

---

# src Documentation

*Release 1.9.15*

**Author**

**Jul 17, 2020**



<b>1</b>	<b>Installation</b>	<b>3</b>
<b>2</b>	<b>Getting started</b>	<b>5</b>
<b>3</b>	<b>Features</b>	<b>11</b>
<b>4</b>	<b>Indices and tables</b>	<b>179</b>
	<b>Python Module Index</b>	<b>181</b>
	<b>Index</b>	<b>183</b>



*psd-tools* is a Python package for working with Adobe Photoshop PSD files as described in [specification](#).



# CHAPTER 1

---

## Installation

---

Use *pip* to install the package:

```
pip install psd-tools
```

---

**Note:** In order to extract images from 32bit PSD files PIL/Pillow must be built with LITTLECMS or LITTLECMS2 support (apt-get install liblcms2-2 or brew install little-cms2)

---





## CHAPTER 2

---

### Getting started

---

```
from psd_tools import PSDImage

psd = PSDImage.open('example.psd')
psd.composite().save('example.png')

for layer in psd:
    print(layer)
    image = layer.composite()
```

Check out the *Usage* documentation for more examples.

## 2.1 Usage

### 2.1.1 Command line

The package provides command line tools to handle a PSD document:

```
psd-tools export <input_file> <output_file> [options]
psd-tools show <input_file> [options]
psd-tools debug <input_file> [options]
psd-tools -h | --help
psd-tools --version
```

Example:

```
psd-tools show example.psd # Show the file content
psd-tools export example.psd example.png # Export as PNG
psd-tools export example.psd[0] example-0.png # Export layer as PNG
```

## 2.1.2 Working with PSD document

`psd_tools.api` package provides the user-friendly API to work with PSD files. *PSDImage* represents a PSD file.

Open an image:

```
from psd_tools import PSDImage
psd = PSDImage.open('my_image.psd')
```

Most of the data structure in the `psd-tools` supports pretty printing in IPython environment.

```
In [1]: PSDImage.open('example.psd')
Out[1]:
PSDImage(mode=RGB size=101x55 depth=8 channels=3)
  [0] PixelLayer('Background' size=101x55)
  [1] PixelLayer('Layer 1' size=85x46)
```

Internal layers are accessible by iterator or indexing:

```
for layer in psd:
    print(layer)
    if layer.is_group():
        for child in layer:
            print(child)

child = psd[0][0]
```

---

**Note:** The iteration order is from background to foreground, which is reversed from version prior to 1.7.x. Use `reversed(list(psd))` to iterate from foreground to background.

---

The opened PSD file can be saved:

```
psd.save('output.psd')
```

## 2.1.3 Working with Layers

There are various layer kinds in Photoshop.

The most basic layer type is *PixelLayer*:

```
print(layer.name)
layer.kind == 'pixel'
```

Some of the layer attributes are editable, such as a layer name:

```
layer.name = 'Updated layer 1'
```

---

**Note:** Currently, the package does not support adding or removing of a layer.

---

*Group* has internal layers:

```
for layer in group:
    print(layer)

first_layer = group[0]
```

*TypeLayer* is a layer with texts:

```
print(layer.text)
```

*ShapeLayer* draws a vector shape, and the shape information is stored in *vector\_mask* and *origination* property. Other layers can also have shape information as a mask:

```
print(layer.vector_mask)
for shape in layer.origination:
    print(shape)
```

*SmartObjectLayer* embeds or links an external file for non-destructive editing. The file content is accessible via *smart\_object* property:

```
import io
if layer.smart_object.filetype in ('jpg', 'png'):
    image = Image.open(io.BytesIO(layer.smart_object.data))
```

*SolidColorFill*, *PatternFill*, and *GradientFill* are fill layers that paint the entire region if there is no associated mask. Sub-classes of *AdjustmentLayer* represents layer adjustment applied to the composed image. See *Adjustment layers*.

## 2.1.4 Exporting data to PIL

Export the entire document as `PIL.Image`:

```
image = psd.composite()
image.save('exported.png')
```

Export a single layer including masks and clipping layers:

```
image = layer.composite()
```

Export layer and mask separately without composition:

```
image = layer.topil()
mask = layer.mask.topil()
```

To composite specific layers, such as layers except for texts, use *layer\_filter* option:

```
image = psd.composite(
    layer_filter=lambda layer: layer.is_visible() and layer.kind != 'type')
```

Note that most of the layer effects and adjustment layers are not supported. The compositing result may look different from Photoshop.

## 2.1.5 Exporting data to NumPy

PSDImage or layers can be exported to NumPy array by *numpy()* method:

```
image = psd.numpy()
layer_image = layer.numpy()
```

## 2.2 Migrating to 1.9

psd-tools 1.9 switches to NumPy based compositing.

version 1.8.x:

```
psd = PSDImage.open(filename)
image = psd.compose()
layer = psd[0]
layer_image = layer.compose()
```

version 1.9.x:

```
psd = PSDImage.open(filename)
image = psd.composite()
layer = psd[0]
layer_image = layer.composite()
```

NumPy array API is introduced:

```
image = psd.numpy()
layer_image = layer.numpy()
```

## 2.3 Migrating to 1.8

There are major API changes in version 1.8.x.

---

**Note:** In version 1.8.0 - 1.8.7, the package name was *psd\_tools2*.

---

### 2.3.1 PSDImage

File open method is changed from *load* to *open()*.

version 1.7.x:

```
psd = PSDImage.load(filename)
with open(filename, 'rb') as f:
    psd = PSDImage.from_stream(f)
```

version 1.8.x:

```
psd = PSDImage.open(filename)
with open(filename, 'rb') as f:
    psd = PSDImage.open(f)
```

## 2.3.2 Layers

Children of PSDImage or Group is directly accessible by iterator or indexing.

version 1.7.x:

```
for layer in group.layers:
    print(layer)

first_child = group.layers[0]
```

version 1.8.x:

```
for layer in group:
    print(layer)

first_child = group[0]
```

In version 1.8.x, the order of layers is reversed to reflect that the index should not change when a new layer is added on top.

## 2.3.3 PIL export

Primary PIL export method is `compose()`.

version 1.7.x:

```
image = psd.as_PIL()

layer_image = compose(layer)
raw_layer_image = layer.as_PIL()
```

version 1.8.x:

```
image = psd.compose()

layer_image = layer.compose()
raw_layer_image = layer.topil()
```

## 2.3.4 Low-level data structure

Data structures are completely rewritten to support writing functionality. See `psd_tools.psd` subpackage.

version 1.7.x:

```
psd.decoded_data
```

version 1.8.x:

```
psd._record
```

## 2.3.5 Drop pymaging support

Pymaging support is dropped.

## 2.4 Contributing

Development happens at github: [bug tracker](#). Feel free to submit [bug reports](#) or pull requests. Attaching an erroneous PSD file makes the debugging process faster. Such PSD file might be added to the test suite.

The license is MIT.

### 2.4.1 Package design

The package consists of two major subpackages:

- 1) `psd_tools.psd`: **subpackage that reads/writes low-level binary** structure of the PSD/PSB file. The core data structures are built around `attrs` class that all implement `read` and `write` methods. Each data object tries to resemble the structure described in the [specification](#). Although documented, the [specification](#) is far from complete and some are even inaccurate. When `psd-tools` finds unknown data structure, the package keeps such data as `bytes` in the parsed result.
- 2) `psd_tools.api`: **User-facing API that implements various** easy-to-use methods that manipulate low-level `psd_tools.psd` data structures.

### 2.4.2 Testing

In order to run tests, make sure PIL/Pillow is built with LittleCMS or LittleCMS2 support, install `tox` and type:

```
tox
```

from the source checkout. Or, it is a good idea to install and run `detox` for parallel execution:

```
detox
```

### 2.4.3 Documentation

Install Sphinx to generate documents:

```
pip install sphinx sphinx_rtd_theme
```

Once installed, use *Makefile*:

```
make docs
```

### 2.4.4 Acknowledgments

Great thanks to [all the contributors](#).

### Supported:

- Read and write of the low-level PSD/PSB file structure;
- Raw layer image export in NumPy and PIL format.

### Limited support:

- Composition of basic pixel-based layers by normal blending;
- Composition of fill layer effects;
- Vector masks;
- Editing of some layer attributes such as layer name;
- Blending modes except for dissolve;
- Drawing of bezier curves.

### Not supported:

- Editing of layer structure, such as adding or removing a layer;
- Composition of adjustment layers;
- Composition of layer effects;
- Font rendering.

## 3.1 psd\_tools

See *Usage* for examples.

### 3.1.1 PSDImage

**class** `psd_tools.PSDImage` (*data*)

Photoshop PSD/PSB file object.

The low-level data structure is accessible at `PSDImage._record`.

Example:

```
from psd_tools import PSDImage

psd = PSDImage.open('example.psd')
image = psd.compose()

for layer in psd:
    layer_image = layer.compose()
```

**bbox**

Minimal bounding box that contains all the visible layers.

Use `viewbox` to get viewport bounding box. When the psd is empty, `bbox` is equal to the canvas bounding box.

**Returns** (left, top, right, bottom) *tuple*.

**bottom**

Bottom coordinate.

**Returns** *int*

**channels**

Number of color channels.

**Returns** *int*

**color\_mode**

Document color mode, such as 'RGB' or 'GRAYSCALE'. See *ColorMode*.

**Returns** *ColorMode*

**compose** (*force=False, bbox=None, layer\_filter=None*)

Deprecated, use *composite()*.

Compose the PSD image.

**Parameters** **bbox** – Viewport tuple (left, top, right, bottom).

**Returns** `PIL.Image`, or `None` if there is no pixel.

**composite** (*viewport=None, force=False, color=1.0, alpha=0.0, layer\_filter=None, ignore\_preview=False*)

Composite the PSD image.

**Parameters**

- **viewport** – Viewport bounding box specified by (x1, y1, x2, y2) tuple. Default is the viewbox of the PSD.
- **ignore\_preview** – Boolean flag to whether skip compositing when a pre-composited preview is available.
- **force** – Boolean flag to force vector drawing.
- **color** – Backdrop color specified by scalar or tuple of scalar. The color value should be in [0.0, 1.0]. For example, (1., 0., 0.) specifies red in RGB color mode.



- **alpha** – Backdrop alpha in [0.0, 1.0].
- **layer\_filter** – Callable that takes a layer as argument and returns whether if the layer is composited. Default is `is_visible()`.

**Returns** `PIL.Image`.

### **depth**

Pixel depth bits.

**Returns** `int`

### **descendants** (*include\_clip=True*)

Return a generator to iterate over all descendant layers.

Example:

```
# Iterate over all layers
for layer in psd.descendants():
    print(layer)

# Iterate over all layers in reverse order
for layer in reversed(list(psd.descendants())):
    print(layer)
```

**Parameters** `include_clip` – include clipping layers.

### **classmethod frompil** (*image, compression=<Compression.RLE: 1>*)

Create a new PSD document from PIL Image.

#### **Parameters**

- **image** – PIL Image object.
- **compression** – ImageData compression option. See *Compression*.

**Returns** A `PSDImage` object.

### **has\_preview** ()

Returns if the document has real merged data. When True, `topil()` returns pre-composed data.

### **has\_thumbnail** ()

True if the `PSDImage` has a thumbnail resource.

### **height**

Document height.

**Returns** `int`

### **image\_resources**

Document image resources. `ImageResources` is a dict-like structure that keeps various document settings.

See `psd_tools.constants.Resource` for available keys.

**Returns** `ImageResources`

Example:

```
from psd_tools.constants import Resource
version_info = psd.image_resources.get_data(Resource.VERSION_INFO)
slices = psd.image_resources.get_data(Resource.SLICES)
```

Image resources contain an ICC profile. The following shows how to export a PNG file with embedded ICC profile:

```
from psd_tools.constants import Resource
icc_profile = psd.image_resources.get_data(Resource.ICC_PROFILE)
image = psd.compose(apply_icc=False)
image.save('output.png', icc_profile=icc_profile)
```

**is\_group()**

Return True if the layer is a group.

**Returns** *bool*

**is\_visible()**

Returns visibility of the element.

**Returns** *bool*

**kind**

Kind.

**Returns** *'psdimage'*

**left**

Left coordinate.

**Returns** *0*

**name**

Element name.

**Returns** *'Root'*

**classmethod new** (*mode, size, color=0, depth=8, \*\*kwargs*)

Create a new PSD document.

**Parameters**

- **mode** – The color mode to use for the new image.
- **size** – A tuple containing (width, height) in pixels.
- **color** – What color to use for the image. Default is black.

**Returns** A PSDImage object.

**numpy** (*channel=None*)

Get NumPy array of the layer.

**Parameters** **channel** – Which channel to return, can be 'color', 'shape', 'alpha', or 'mask'.  
Default is 'color+alpha'.

**Returns** *numpy.ndarray*

**offset**

(left, top) tuple.

**Returns** *tuple*

**classmethod open** (*fp, \*\*kwargs*)

Open a PSD document.

**Parameters**

- **fp** – filename or file-like object.

- **encoding** – charset encoding of the pascal string within the file, default ‘macroman’. Some psd files need explicit encoding option.

**Returns** A PSDImage object.

**parent**

Parent of this layer.

**right**

Right coordinate.

**Returns** *int*

**save** (*fp*, *mode*=‘wb’, *\*\*kwargs*)

Save the PSD file.

**Parameters**

- **fp** – filename or file-like object.
- **encoding** – charset encoding of the pascal string within the file, default ‘macroman’.
- **mode** – file open mode, default ‘wb’.

**size**

(width, height) tuple.

**Returns** *tuple*

**tagged\_blocks**

Document tagged blocks that is a dict-like container of settings.

See `psd_tools.constants.Tag` for available keys.

**Returns** *TaggedBlocks* or *None*.

Example:

```
from psd_tools.constants import Tag
patterns = psd.tagged_blocks.get_data(Tag.PATTERNS1)
```

**thumbnail** ()

Returns a thumbnail image in PIL.Image. When the file does not contain an embedded thumbnail image, returns None.

**top**

Top coordinate.

**Returns** *0*

**topil** (*channel*=None, *apply\_icc*=False)

Get PIL Image.

**Parameters**

- **channel** – Which channel to return; e.g., 0 for ‘R’ channel in RGB image. See `ChannelID`. When *None*, the method returns all the channels supported by PIL modes.
- **apply\_icc** – Whether to apply ICC profile conversion to sRGB.

**Returns** `PIL.Image`, or *None* if the composed image is not available.

**version**

Document version. PSD file is 1, and PSB file is 2.

**Returns** *int*

**viewbox**

Return bounding box of the viewport.

**Returns** (left, top, right, bottom) *tuple*.

**visible**

Visibility.

**Returns** *True*

**width**

Document width.

**Returns** *int*

### 3.1.2 compose

`psd_tools.compose` (*layers, force=False, bbox=None, layer\_filter=None, context=None, color=None*)  
Compose layers to a single `PIL.Image`. If the layers do not have visible pixels, the function returns *None*.

Example:

```
image = compose([layer1, layer2])
```

In order to skip some layers, pass *layer\_filter* function which should take *layer* as an argument and return *True* to keep the layer or return *False* to skip:

```
image = compose(  
    layers,  
    layer_filter=lambda x: x.is_visible() and x.kind == 'type'  
)
```

By default, visible layers are composed.

---

**Note:** This function is experimental and does not guarantee Photoshop-quality rendering.

Currently the following are ignored:

- Adjustments layers
- Layer effects
- Blending modes: dissolve and darker/lighter color becomes normal

Shape drawing is inaccurate if the PSD file is not saved with maximum compatibility.

Some of the blending modes do not reproduce photoshop blending.

---

#### Parameters

- **layers** – a layer, or an iterable of layers.
- **bbox** – (left, top, bottom, right) tuple that specifies a region to compose. By default, all the visible area is composed. The origin is at the top-left corner of the PSD document.
- **context** – *PIL.Image* object for the backdrop rendering context. Must be used with the correct *bbox* size.
- **layer\_filter** – a callable that takes a layer and returns *bool*.
- **color** – background color in *int* or *tuple*.

- **kwargs** – arguments passed to underlying *topil()* call.

**Returns** `PIL.Image` or `None`.

## 3.2 psd\_tools.api.adjustments

Adjustment and fill layers.

Example:

```
if layer.kind == 'brightnesscontrast':
    print(layer.brightness)

if layer.kind == 'gradientfill':
    print(layer.gradient_kind)
```

### 3.2.1 Fill layers

Fill layers are similar to *ShapeLayer* except that the layer might not have an associated vector mask. The layer therefore expands the entire canvas of the PSD document.

Fill layers all inherit from `FillLayer`.

Example:

```
if isinstance(layer, psd_tools.layers.FillLayer):
    image = layer.compose()
```

#### SolidColorFill

**class** `psd_tools.api.adjustments.SolidColorFill(*args)`  
Solid color fill.

**bbox**  
(left, top, right, bottom) tuple.

**blend\_mode**  
Blend mode of this layer. Writable.

Example:

```
from psd_tools.constants import BlendMode
if layer.blend_mode == BlendMode.NORMAL:
    layer.blend_mode = BlendMode.SCREEN
```

**Returns** `BlendMode`.

**bottom**  
Bottom coordinate.

**Returns** `int`

**clip\_layers**  
Clip layers associated with this layer.

To compose clipping layers:

```
from psd_tools import compose
clip_mask = compose(layer.clip_layers)
```

**Returns** list of layers

**compose** (*force=False, bbox=None, layer\_filter=None*)

Deprecated, use *composite()*.

Compose layer and masks (mask, vector mask, and clipping layers).

Note that the resulting image size is not necessarily equal to the layer size due to different mask dimensions. The offset of the composed image is stored at *.info['offset']* attribute of *PIL.Image*.

**Parameters** **bbox** – Viewport bounding box specified by (x1, y1, x2, y2) tuple.

**Returns** *PIL.Image*, or *None* if the layer has no pixel.

**composite** (*viewport=None, force=False, color=1.0, alpha=0.0, layer\_filter=None*)

Composite layer and masks (mask, vector mask, and clipping layers).

**Parameters**

- **viewport** – Viewport bounding box specified by (x1, y1, x2, y2) tuple. Default is the layer's bbox.
- **force** – Boolean flag to force vector drawing.
- **color** – Backdrop color specified by scalar or tuple of scalar. The color value should be in [0.0, 1.0]. For example, (1., 0., 0.) specifies red in RGB color mode.
- **alpha** – Backdrop alpha in [0.0, 1.0].
- **layer\_filter** – Callable that takes a layer as argument and returns whether if the layer is composited. Default is *is\_visible()*.

**Returns** *PIL.Image*.

**data**

Color in Descriptor(RGB).

**effects**

Layer effects.

**Returns** *Effects*

**has\_clip\_layers** ()

Returns True if the layer has associated clipping.

**Returns** *bool*

**has\_effects** ()

Returns True if the layer has effects.

**Returns** *bool*

**has\_mask** ()

Returns True if the layer has a mask.

**Returns** *bool*

**has\_origination** ()

Returns True if the layer has live shape properties.

**Returns** *bool*

**has\_pixels()**

Returns True if the layer has associated pixels. When this is True, *topil* method returns `PIL.Image`.

**Returns** *bool*

**has\_stroke()**

Returns True if the shape has a stroke.

**has\_vector\_mask()**

Returns True if the layer has a vector mask.

**Returns** *bool*

**height**

Height of the layer.

**Returns** *int*

**is\_group()**

Return True if the layer is a group.

**Returns** *bool*

**is\_visible()**

Layer visibility. Takes group visibility in account.

**Returns** *bool*

**kind**

Kind of this layer, such as group, pixel, shape, type, smartobject, or psdimage. Class name without *layer* suffix.

**Returns** *str*

**layer\_id**

Layer ID.

**Returns** *int* layer id. if the layer is not assigned an id, -1.

**left**

Left coordinate. Writable.

**Returns** *int*

**mask**

Returns mask associated with this layer.

**Returns** *Mask* or *None*

**name**

Layer name. Writable.

**Returns** *str*

**numpy** (*channel=None*)

Get NumPy array of the layer.

**Parameters** **channel** – Which channel to return, can be ‘color’, ‘shape’, ‘alpha’, or ‘mask’.  
Default is ‘color+alpha’.

**Returns** `numpy.ndarray` or *None* if there is no pixel.

**offset**

(left, top) tuple. Writable.

**Returns** *tuple*

**opacity**

Opacity of this layer in [0, 255] range. Writable.

**Returns** int

**origination**

Property for a list of live shapes or a line.

Some of the vector masks have associated live shape properties, that are Photoshop feature to handle primitive shapes such as a rectangle, an ellipse, or a line. Vector masks without live shape properties are plain path objects.

See `psd_tools.api.shape`.

**Returns** List of *Invalidated*, *Rectangle*, *RoundedRectangle*, *Ellipse*, or *Line*.

**parent**

Parent of this layer.

**right**

Right coordinate.

**Returns** int

**size**

(width, height) tuple.

**Returns** *tuple*

**stroke**

Property for strokes.

**tagged\_blocks**

Layer tagged blocks that is a dict-like container of settings.

See `psd_tools.constants.Tag` for available keys.

**Returns** *TaggedBlocks* or *None*.

Example:

```
from psd_tools.constants import Tag
metadata = layer.tagged_blocks.get_data(Tag.METADATA_SETTING)
```

**top**

Top coordinate. Writable.

**Returns** int

**topil** (*channel=None*, *apply\_icc=False*)

Get PIL Image of the layer.

**Parameters**

- **channel** – Which channel to return; e.g., 0 for ‘R’ channel in RGB image. See *ChannelID*. When *None*, the method returns all the channels supported by PIL modes.
- **apply\_icc** – Whether to apply ICC profile conversion to sRGB.

**Returns** *PIL*.Image, or *None* if the layer has no pixels.

Example:



```

from psd_tools.constants import ChannelID

image = layer.topil()
red = layer.topil(ChannelID.CHANNEL_0)
alpha = layer.topil(ChannelID.TRANSPARENCY_MASK)

```

---

**Note:** Not all of the PSD image modes are supported in `PIL.Image`. For example, ‘CMYK’ mode cannot include alpha channel in PIL. In this case, `topil` drops alpha channel.

---

**vector\_mask**

Returns vector mask associated with this layer.

**Returns** *VectorMask* or *None*

**visible**

Layer visibility. Doesn’t take group visibility in account. Writable.

**Returns** *bool*

**width**

Width of the layer.

**Returns** *int*

**PatternFill**

**class** `psd_tools.api.adjustments.PatternFill(*args)`  
 Pattern fill.

**bbox**

(left, top, right, bottom) tuple.

**blend\_mode**

Blend mode of this layer. Writable.

Example:

```

from psd_tools.constants import BlendMode
if layer.blend_mode == BlendMode.NORMAL:
    layer.blend_mode = BlendMode.SCREEN

```

**Returns** *BlendMode*.

**bottom**

Bottom coordinate.

**Returns** *int*

**clip\_layers**

Clip layers associated with this layer.

To compose clipping layers:

```

from psd_tools import compose
clip_mask = compose(layer.clip_layers)

```

**Returns** list of layers

**compose** (*force=False, bbox=None, layer\_filter=None*)

Deprecated, use `composite()`.

Compose layer and masks (mask, vector mask, and clipping layers).

Note that the resulting image size is not necessarily equal to the layer size due to different mask dimensions. The offset of the composed image is stored at `.info['offset']` attribute of `PIL.Image`.

**Parameters** **bbox** – Viewport bounding box specified by (x1, y1, x2, y2) tuple.

**Returns** `PIL.Image`, or `None` if the layer has no pixel.

**composite** (*viewport=None, force=False, color=1.0, alpha=0.0, layer\_filter=None*)

Composite layer and masks (mask, vector mask, and clipping layers).

**Parameters**

- **viewport** – Viewport bounding box specified by (x1, y1, x2, y2) tuple. Default is the layer's `bbox`.
- **force** – Boolean flag to force vector drawing.
- **color** – Backdrop color specified by scalar or tuple of scalar. The color value should be in [0.0, 1.0]. For example, (1., 0., 0.) specifies red in RGB color mode.
- **alpha** – Backdrop alpha in [0.0, 1.0].
- **layer\_filter** – Callable that takes a layer as argument and returns whether if the layer is composited. Default is `is_visible()`.

**Returns** `PIL.Image`.

**data**

Pattern in Descriptor(PATTERN).

**effects**

Layer effects.

**Returns** `Effects`

**has\_clip\_layers** ()

Returns True if the layer has associated clipping.

**Returns** `bool`

**has\_effects** ()

Returns True if the layer has effects.

**Returns** `bool`

**has\_mask** ()

Returns True if the layer has a mask.

**Returns** `bool`

**has\_origination** ()

Returns True if the layer has live shape properties.

**Returns** `bool`

**has\_pixels** ()

Returns True if the layer has associated pixels. When this is True, `topil` method returns `PIL.Image`.

**Returns** `bool`

**has\_stroke** ()

Returns True if the shape has a stroke.

**has\_vector\_mask()**

Returns True if the layer has a vector mask.

**Returns** *bool*

**height**

Height of the layer.

**Returns** *int*

**is\_group()**

Return True if the layer is a group.

**Returns** *bool*

**is\_visible()**

Layer visibility. Takes group visibility in account.

**Returns** *bool*

**kind**

Kind of this layer, such as group, pixel, shape, type, smartobject, or psdimage. Class name without *layer* suffix.

**Returns** *str*

**layer\_id**

Layer ID.

**Returns** *int* layer id. if the layer is not assigned an id, -1.

**left**

Left coordinate. Writable.

**Returns** *int*

**mask**

Returns mask associated with this layer.

**Returns** *Mask* or *None*

**name**

Layer name. Writable.

**Returns** *str*

**numpy** (*channel=None*)

Get NumPy array of the layer.

**Parameters** **channel** – Which channel to return, can be ‘color’, ‘shape’, ‘alpha’, or ‘mask’.  
Default is ‘color+alpha’.

**Returns** *numpy.ndarray* or *None* if there is no pixel.

**offset**

(left, top) tuple. Writable.

**Returns** *tuple*

**opacity**

Opacity of this layer in [0, 255] range. Writable.

**Returns** *int*

**origination**

Property for a list of live shapes or a line.

Some of the vector masks have associated live shape properties, that are Photoshop feature to handle primitive shapes such as a rectangle, an ellipse, or a line. Vector masks without live shape properties are plain path objects.

See `psd_tools.api.shape`.

**Returns** List of *Invalidated*, *Rectangle*, *RoundedRectangle*, *Ellipse*, or *Line*.

**parent**

Parent of this layer.

**right**

Right coordinate.

**Returns** int

**size**

(width, height) tuple.

**Returns** tuple

**stroke**

Property for strokes.

**tagged\_blocks**

Layer tagged blocks that is a dict-like container of settings.

See `psd_tools.constants.Tag` for available keys.

**Returns** *TaggedBlocks* or *None*.

Example:

```
from psd_tools.constants import Tag
metadata = layer.tagged_blocks.get_data(Tag.METADATA_SETTING)
```

**top**

Top coordinate. Writable.

**Returns** int

**topil** (*channel=None*, *apply\_icc=False*)

Get PIL Image of the layer.

**Parameters**

- **channel** – Which channel to return; e.g., 0 for ‘R’ channel in RGB image. See *ChannelID*. When *None*, the method returns all the channels supported by PIL modes.
- **apply\_icc** – Whether to apply ICC profile conversion to sRGB.

**Returns** PIL . Image, or *None* if the layer has no pixels.

Example:

```
from psd_tools.constants import ChannelID

image = layer.topil()
red = layer.topil(ChannelID.CHANNEL_0)
alpha = layer.topil(ChannelID.TRANSPARENCY_MASK)
```

---

**Note:** Not all of the PSD image modes are supported in `PIL.Image`. For example, ‘CMYK’ mode cannot include alpha channel in PIL. In this case, `topil` drops alpha channel.

---

**vector\_mask**

Returns vector mask associated with this layer.

**Returns** `VectorMask` or `None`

**visible**

Layer visibility. Doesn’t take group visibility in account. Writable.

**Returns** `bool`

**width**

Width of the layer.

**Returns** `int`

**GradientFill**

**class** `psd_tools.api.adjustments.GradientFill(*args)`

Gradient fill.

**bbox**

(left, top, right, bottom) tuple.

**blend\_mode**

Blend mode of this layer. Writable.

Example:

```
from psd_tools.constants import BlendMode
if layer.blend_mode == BlendMode.NORMAL:
    layer.blend_mode = BlendMode.SCREEN
```

**Returns** `BlendMode`.

**bottom**

Bottom coordinate.

**Returns** `int`

**clip\_layers**

Clip layers associated with this layer.

To compose clipping layers:

```
from psd_tools import compose
clip_mask = compose(layer.clip_layers)
```

**Returns** list of layers

**compose** (*force=False, bbox=None, layer\_filter=None*)

Deprecated, use `composite()`.

Compose layer and masks (mask, vector mask, and clipping layers).

Note that the resulting image size is not necessarily equal to the layer size due to different mask dimensions. The offset of the composed image is stored at `.info['offset']` attribute of `PIL.Image`.

**Parameters** `bbox` – Viewport bounding box specified by (x1, y1, x2, y2) tuple.

**Returns** `PIL.Image`, or `None` if the layer has no pixel.

**composite** (`viewport=None, force=False, color=1.0, alpha=0.0, layer_filter=None`)

Composite layer and masks (mask, vector mask, and clipping layers).

**Parameters**

- **viewport** – Viewport bounding box specified by (x1, y1, x2, y2) tuple. Default is the layer's `bbox`.
- **force** – Boolean flag to force vector drawing.
- **color** – Backdrop color specified by scalar or tuple of scalar. The color value should be in [0.0, 1.0]. For example, (1., 0., 0.) specifies red in RGB color mode.
- **alpha** – Backdrop alpha in [0.0, 1.0].
- **layer\_filter** – Callable that takes a layer as argument and returns whether if the layer is composited. Default is `is_visible()`.

**Returns** `PIL.Image`.

**data**

Gradient in Descriptor(`GRADIENT`).

**effects**

Layer effects.

**Returns** `Effects`

**gradient\_kind**

Kind of the gradient, one of the following:

- *Linear*
- *Radial*
- *Angle*
- *Reflected*
- *Diamond*

**has\_clip\_layers** ()

Returns True if the layer has associated clipping.

**Returns** `bool`

**has\_effects** ()

Returns True if the layer has effects.

**Returns** `bool`

**has\_mask** ()

Returns True if the layer has a mask.

**Returns** `bool`

**has\_origination** ()

Returns True if the layer has live shape properties.

**Returns** `bool`

**has\_pixels()**

Returns True if the layer has associated pixels. When this is True, *topil* method returns `PIL.Image`.

**Returns** *bool*

**has\_stroke()**

Returns True if the shape has a stroke.

**has\_vector\_mask()**

Returns True if the layer has a vector mask.

**Returns** *bool*

**height**

Height of the layer.

**Returns** *int*

**is\_group()**

Return True if the layer is a group.

**Returns** *bool*

**is\_visible()**

Layer visibility. Takes group visibility in account.

**Returns** *bool*

**kind**

Kind of this layer, such as group, pixel, shape, type, smartobject, or psdimage. Class name without *layer* suffix.

**Returns** *str*

**layer\_id**

Layer ID.

**Returns** *int* layer id. if the layer is not assigned an id, -1.

**left**

Left coordinate. Writable.

**Returns** *int*

**mask**

Returns mask associated with this layer.

**Returns** *Mask* or *None*

**name**

Layer name. Writable.

**Returns** *str*

**numpy (channel=None)**

Get NumPy array of the layer.

**Parameters** **channel** – Which channel to return, can be ‘color’, ‘shape’, ‘alpha’, or ‘mask’.  
Default is ‘color+alpha’.

**Returns** `numpy.ndarray` or *None* if there is no pixel.

**offset**

(left, top) tuple. Writable.

**Returns** *tuple*

**opacity**

Opacity of this layer in [0, 255] range. Writable.

**Returns** int

**origination**

Property for a list of live shapes or a line.

Some of the vector masks have associated live shape properties, that are Photoshop feature to handle primitive shapes such as a rectangle, an ellipse, or a line. Vector masks without live shape properties are plain path objects.

See `psd_tools.api.shape`.

**Returns** List of *Invalidated*, *Rectangle*, *RoundedRectangle*, *Ellipse*, or *Line*.

**parent**

Parent of this layer.

**right**

Right coordinate.

**Returns** int

**size**

(width, height) tuple.

**Returns** *tuple*

**stroke**

Property for strokes.

**tagged\_blocks**

Layer tagged blocks that is a dict-like container of settings.

See `psd_tools.constants.Tag` for available keys.

**Returns** *TaggedBlocks* or *None*.

Example:

```
from psd_tools.constants import Tag
metadata = layer.tagged_blocks.get_data(Tag.METADATA_SETTING)
```

**top**

Top coordinate. Writable.

**Returns** int

**topil** (*channel=None*, *apply\_icc=False*)

Get PIL Image of the layer.

**Parameters**

- **channel** – Which channel to return; e.g., 0 for ‘R’ channel in RGB image. See *ChannelID*. When *None*, the method returns all the channels supported by PIL modes.
- **apply\_icc** – Whether to apply ICC profile conversion to sRGB.

**Returns** *PIL*.Image, or *None* if the layer has no pixels.

Example:



```

from psd_tools.constants import ChannelID

image = layer.topil()
red = layer.topil(ChannelID.CHANNEL_0)
alpha = layer.topil(ChannelID.TRANSPARENCY_MASK)

```

---

**Note:** Not all of the PSD image modes are supported in `PIL.Image`. For example, ‘CMYK’ mode cannot include alpha channel in PIL. In this case, `topil` drops alpha channel.

---

**vector\_mask**

Returns vector mask associated with this layer.

**Returns** *VectorMask* or *None*

**visible**

Layer visibility. Doesn’t take group visibility in account. Writable.

**Returns** *bool*

**width**

Width of the layer.

**Returns** *int*

### 3.2.2 Adjustment layers

Adjustment layers apply image filtering to the composed result. All adjustment layers inherit from `AdjustmentLayer`. Adjustment layers do not have pixels, and currently ignored in `compose`. Attempts to call `topil` on adjustment layers always return *None*.

Just as any other layer, adjustment layers might have an associated mask or vector mask. Adjustment can appear in other layers’ clipping layers.

Example:

```

if isinstance(layer, psd_tools.layers.AdjustmentLayer):
    print(layer.kind)

```

### BrightnessContrast

**class** `psd_tools.api.adjustments.BrightnessContrast` (\*args)

Brightness and contrast adjustment.

**automatic**

**brightness**

**contrast**

**lab**

**mean**

**use\_legacy**

**vrsn**

## Curves

```
class psd_tools.api.adjustments.Curves(*args)
    Curves adjustment.

    data
        Raw data.

        Returns Curves

    extra
```

## Exposure

```
class psd_tools.api.adjustments.Exposure(*args)
    Exposure adjustment.

    exposure
        Exposure.

        Returns float

    gamma
        Gamma.

        Returns float

    offset
        Offset.

        Returns float
```

## Levels

```
class psd_tools.api.adjustments.Levels(*args)
    Levels adjustment.

    Levels contain a list of LevelRecord.

    data
        List of level records. The first record is the master.

        Returns Levels.

    master
        Master record.
```

## Vibrance

```
class psd_tools.api.adjustments.Vibrance(*args)
    Vibrance adjustment.

    saturation
        Saturation.

        Returns int

    vibrance
        Vibrance.
```

**Returns** *int*

## HueSaturation

**class** `psd_tools.api.adjustments.HueSaturation(*args)`  
Hue/Saturation adjustment.

HueSaturation contains a list of data.

**colorization**  
Colorization.

**Returns** *tuple*

**data**  
List of Hue/Saturation records.

**Returns** *list*

**enable\_colorization**  
Enable colorization.

**Returns** *int*

**master**  
Master record.

**Returns** *tuple*

## ColorBalance

**class** `psd_tools.api.adjustments.ColorBalance(*args)`  
Color balance adjustment.

**highlights**  
Highlights.

**Returns** *tuple*

**luminosity**  
Luminosity.

**Returns** *int*

**midtone**  
Mid-tones.

**Returns** *tuple*

**shadows**  
Shadows.

**Returns** *tuple*

## BlackAndWhite

**class** `psd_tools.api.adjustments.BlackAndWhite(*args)`  
Black and white adjustment.

**blue**

**cyan**  
**green**  
**magenta**  
**preset\_file\_name**  
**preset\_kind**  
**red**  
**tint\_color**  
**use\_tint**  
**yellow**

### PhotoFilter

```
class psd_tools.api.adjustments.PhotoFilter(*args)
    Photo filter adjustment.

    color_components
    color_space
    density
    luminosity
    xyz
        xyz.

        Returns bool
```

### ChannelMixer

```
class psd_tools.api.adjustments.ChannelMixer(*args)
    Channel mixer adjustment.

    data
    monochrome
```

### ColorLookup

```
class psd_tools.api.adjustments.ColorLookup(*args)
    Color lookup adjustment.
```

### Posterize

```
class psd_tools.api.adjustments.Posterize(*args)
    Posterize adjustment.

    posterize
        Posterize value.

        Returns int
```

## Threshold

```
class psd_tools.api.adjustments.Threshold(*args)
    Threshold adjustment.

    threshold
        Threshold value.

    Returns int
```

## SelectiveColor

```
class psd_tools.api.adjustments.SelectiveColor(*args)
    Selective color adjustment.

    data

    method
```

## GradientMap

```
class psd_tools.api.adjustments.GradientMap(*args)
    Gradient map adjustment.

    color_model

    color_stops

    dithered

    expansion

    gradient_name

    interpolation
        Interpolation between 0.0 and 1.0.

    length

    max_color

    min_color

    mode

    random_seed

    reversed

    roughness

    show_transparency

    transparency_stops

    use_vector_color
```

## 3.3 psd\_tools.api.effects

Effects module.

**class** `psd_tools.api.effects.Effects` (*layer*)  
List-like effects.

**enabled**  
Whether if all the effects are enabled.

**Return type** `bool`

**find** (*name*)  
Iterate effect items by name.

**scale**  
Scale value.

### 3.3.1 DropShadow

**class** `psd_tools.api.effects.DropShadow` (*value, image\_resources*)

**angle**  
Angle value.

**anti\_aliased**  
Anti-aliased.

**blend\_mode**  
Effect blending mode.

**choke**  
Choke level.

**color**  
Color.

**contour**  
Contour configuration.

**distance**  
Distance.

**enabled**  
Whether if the effect is enabled.

**layer\_knocks\_out**  
Layers are knocking out.

**noise**  
Noise level.

**opacity**  
Layer effect opacity in percentage.

**present**  
Whether if the effect is present in Photoshop UI.

**shown**  
Whether if the effect is shown in dialog.

**size**  
Size in pixels.

**use\_global\_light**  
Using global light.

### 3.3.2 InnerShadow

**class** `psd_tools.api.effects.InnerShadow` (*value, image\_resources*)

**angle**  
Angle value.

**anti\_aliased**  
Angi-aliased.

**blend\_mode**  
Effect blending mode.

**choke**  
Choke level.

**color**  
Color.

**contour**  
Contour configuration.

**distance**  
Distance.

**enabled**  
Whether if the effect is enabled.

**noise**  
Noise level.

**opacity**  
Layer effect opacity in percentage.

**present**  
Whether if the effect is present in Photoshop UI.

**shown**  
Whether if the effect is shown in dialog.

**size**  
Size in pixels.

**use\_global\_light**  
Using global light.

### 3.3.3 OuterGlow

**class** `psd_tools.api.effects.OuterGlow` (*value, image\_resources*)

**anti\_aliased**  
Angi-aliased.

**blend\_mode**  
Effect blending mode.

**choke**  
Choke level.

**color**  
Color.

**contour**  
Contour configuration.

**enabled**  
Whether if the effect is enabled.

**glow\_type**  
Glow type.

**gradient**  
Gradient configuration.

**noise**  
Noise level.

**opacity**  
Layer effect opacity in percentage.

**present**  
Whether if the effect is present in Photoshop UI.

**quality\_jitter**  
Quality jitter

**quality\_range**  
Quality range.

**shown**  
Whether if the effect is shown in dialog.

**size**  
Size in pixels.

### 3.3.4 InnerGlow

**class** `psd_tools.api.effects.InnerGlow` (*value, image\_resources*)

**anti\_aliased**  
Angi-aliased.

**blend\_mode**  
Effect blending mode.

**choke**  
Choke level.

**color**  
Color.

**contour**  
Contour configuration.



**enabled**  
Whether if the effect is enabled.

**glow\_source**  
Elements source.

**glow\_type**  
Glow type.

**gradient**  
Gradient configuration.

**noise**  
Noise level.

**opacity**  
Layer effect opacity in percentage.

**present**  
Whether if the effect is present in Photoshop UI.

**quality\_jitter**  
Quality jitter

**quality\_range**  
Quality range.

**shown**  
Whether if the effect is shown in dialog.

**size**  
Size in pixels.

### 3.3.5 ColorOverlay

**class** `psd_tools.api.effects.ColorOverlay` (*value, image\_resources*)

**blend\_mode**  
Effect blending mode.

**color**  
Color.

**enabled**  
Whether if the effect is enabled.

**opacity**  
Layer effect opacity in percentage.

**present**  
Whether if the effect is present in Photoshop UI.

**shown**  
Whether if the effect is shown in dialog.

### 3.3.6 GradientOverlay

**class** `psd_tools.api.effects.GradientOverlay` (*value, image\_resources*)

**aligned**  
Aligned.

**angle**  
Angle value.

**blend\_mode**  
Effect blending mode.

**dithered**  
Dither flag.

**enabled**  
Whether if the effect is enabled.

**gradient**  
Gradient configuration.

**offset**  
Offset value.

**opacity**  
Layer effect opacity in percentage.

**present**  
Whether if the effect is present in Photoshop UI.

**reversed**  
Reverse flag.

**scale**  
Scale value.

**shown**  
Whether if the effect is shown in dialog.

**type**  
Gradient type, one of *linear*, *radial*, *angle*, *reflected*, or *diamond*.

### 3.3.7 PatternOverlay

**class** `psd_tools.api.effects.PatternOverlay` (*value*, *image\_resources*)

**aligned**  
Aligned.

**blend\_mode**  
Effect blending mode.

**enabled**  
Whether if the effect is enabled.

**opacity**  
Layer effect opacity in percentage.

**pattern**  
Pattern config.

**phase**  
Phase value in Point.

**present**

Whether if the effect is present in Photoshop UI.

**scale**

Scale value.

**shown**

Whether if the effect is shown in dialog.

### 3.3.8 Stroke

**class** `psd_tools.api.effects.Stroke` (*value, image\_resources*)

**blend\_mode**

Effect blending mode.

**color**

Color.

**enabled**

Whether if the effect is enabled.

**fill\_type**

Fill type, SolidColor, Gradient, or Pattern.

**gradient**

Gradient configuration.

**opacity**

Layer effect opacity in percentage.

**overprint**

Overprint flag.

**pattern**

Pattern config.

**position**

Position of the stroke, InsetFrame, OutsetFrame, or CenteredFrame.

**present**

Whether if the effect is present in Photoshop UI.

**shown**

Whether if the effect is shown in dialog.

**size**

Size value.

### 3.3.9 BevelEmboss

**class** `psd_tools.api.effects.BevelEmboss` (*value, image\_resources*)

**altitude**

Altitude value.

**angle**

Angle value.

**anti\_aliased**

Anti-aliased.

**bevel\_style**

Bevel style, one of *OuterBevel*, *InnerBevel*, *Emboss*, *PillowEmboss*, or *StrokeEmboss*.

**bevel\_type**

Bevel type, one of *SoftMatte*, *HardLight*, *SoftLight*.

**contour**

Contour configuration.

**depth**

Depth value.

**direction**

Direction, either *StampIn* or *StampOut*.

**enabled**

Whether if the effect is enabled.

**highlight\_color**

Highlight color value.

**highlight\_mode**

Highlight blending mode.

**highlight\_opacity**

Highlight opacity value.

**opacity**

Layer effect opacity in percentage.

**present**

Whether if the effect is present in Photoshop UI.

**shadow\_color**

Shadow color value.

**shadow\_mode**

Shadow blending mode.

**shadow\_opacity**

Shadow opacity value.

**shown**

Whether if the effect is shown in dialog.

**size**

Size value in pixel.

**soften**

Soften value.

**use\_global\_light**

Using global light.

**use\_shape**

Using shape.

**use\_texture**

Using texture.

### 3.3.10 Satin

**class** `psd_tools.api.effects.Satin` (*value, image\_resources*)

Satin effect

**angle**

Angle value.

**anti\_aliased**

Anti-aliased.

**blend\_mode**

Effect blending mode.

**color**

Color.

**contour**

Contour configuration.

**distance**

Distance value.

**enabled**

Whether if the effect is enabled.

**inverted**

Inverted.

**opacity**

Layer effect opacity in percentage.

**present**

Whether if the effect is present in Photoshop UI.

**shown**

Whether if the effect is shown in dialog.

**size**

Size value in pixel.

## 3.4 psd\_tools.api.layers

Layer module.

### 3.4.1 Artboard

**class** `psd_tools.api.layers.Artboard` (*\*args*)

Artboard is a special kind of group that has a pre-defined viewbox.

Example:

```
artboard = psd[1]
image = artboard.compose()
```

**bbox**

(left, top, right, bottom) tuple.

**blend\_mode**

Blend mode of this layer. Writable.

Example:

```
from psd_tools.constants import BlendMode
if layer.blend_mode == BlendMode.NORMAL:
    layer.blend_mode = BlendMode.SCREEN
```

**Returns** *BlendMode*.

**bottom**

Bottom coordinate.

**Returns** int

**clip\_layers**

Clip layers associated with this layer.

To compose clipping layers:

```
from psd_tools import compose
clip_mask = compose(layer.clip_layers)
```

**Returns** list of layers

**compose** (*bbox=None, \*\*kwargs*)

Compose the artboard.

See *compose()* for available extra arguments.

**Parameters** **bbox** – Viewport tuple (left, top, right, bottom).

**Returns** PIL. Image, or *None* if there is no pixel.

**composite** (*viewport=None, force=False, color=1.0, alpha=0.0, layer\_filter=None*)

Composite layer and masks (mask, vector mask, and clipping layers).

**Parameters**

- **viewport** – Viewport bounding box specified by (x1, y1, x2, y2) tuple. Default is the layer's bbox.
- **force** – Boolean flag to force vector drawing.
- **color** – Backdrop color specified by scalar or tuple of scalar. The color value should be in [0.0, 1.0]. For example, (1., 0., 0.) specifies red in RGB color mode.
- **alpha** – Backdrop alpha in [0.0, 1.0].
- **layer\_filter** – Callable that takes a layer as argument and returns whether if the layer is composited. Default is *is\_visible()*.

**Returns** PIL. Image.

**descendants** (*include\_clip=True*)

Return a generator to iterate over all descendant layers.

Example:

```

# Iterate over all layers
for layer in psd.descendants():
    print(layer)

# Iterate over all layers in reverse order
for layer in reversed(list(psd.descendants())):
    print(layer)

```

**Parameters** `include_clip` – include clipping layers.

#### **effects**

Layer effects.

**Returns** *Effects*

#### **static extract\_bbox** (*layers*, *include\_invisible=False*)

Returns a bounding box for `layers` or (0, 0, 0, 0) if the layers have no bounding box.

**Parameters** `include_invisible` – include invisible layers in calculation.

**Returns** tuple of four int

#### **has\_clip\_layers** ()

Returns True if the layer has associated clipping.

**Returns** *bool*

#### **has\_effects** ()

Returns True if the layer has effects.

**Returns** *bool*

#### **has\_mask** ()

Returns True if the layer has a mask.

**Returns** *bool*

#### **has\_origination** ()

Returns True if the layer has live shape properties.

**Returns** *bool*

#### **has\_pixels** ()

Returns True if the layer has associated pixels. When this is True, `topil` method returns `PIL.Image`.

**Returns** *bool*

#### **has\_stroke** ()

Returns True if the shape has a stroke.

#### **has\_vector\_mask** ()

Returns True if the layer has a vector mask.

**Returns** *bool*

#### **height**

Height of the layer.

**Returns** *int*

#### **is\_group** ()

Return True if the layer is a group.

**Returns** *bool*

**is\_visible()**

Layer visibility. Takes group visibility in account.

**Returns** *bool*

**kind**

Kind of this layer, such as group, pixel, shape, type, smartobject, or psdimage. Class name without *layer* suffix.

**Returns** *str*

**layer\_id**

Layer ID.

**Returns** int layer id. if the layer is not assigned an id, -1.

**left**

Left coordinate. Writable.

**Returns** int

**mask**

Returns mask associated with this layer.

**Returns** *Mask* or *None*

**name**

Layer name. Writable.

**Returns** *str*

**numpy** (*channel=None*)

Get NumPy array of the layer.

**Parameters** **channel** – Which channel to return, can be ‘color’, ‘shape’, ‘alpha’, or ‘mask’.  
Default is ‘color+alpha’.

**Returns** `numpy.ndarray` or *None* if there is no pixel.

**offset**

(left, top) tuple. Writable.

**Returns** *tuple*

**opacity**

Opacity of this layer in [0, 255] range. Writable.

**Returns** int

**origination**

Property for a list of live shapes or a line.

Some of the vector masks have associated live shape properties, that are Photoshop feature to handle primitive shapes such as a rectangle, an ellipse, or a line. Vector masks without live shape properties are plain path objects.

See `psd_tools.api.shape`.

**Returns** List of *Invalidated*, *Rectangle*, *RoundedRectangle*, *Ellipse*, or *Line*.

**parent**

Parent of this layer.

**right**

Right coordinate.



**Returns** int

**size**

(width, height) tuple.

**Returns** *tuple*

**stroke**

Property for strokes.

**tagged\_blocks**

Layer tagged blocks that is a dict-like container of settings.

See `psd_tools.constants.Tag` for available keys.

**Returns** *TaggedBlocks* or *None*.

Example:

```
from psd_tools.constants import Tag
metadata = layer.tagged_blocks.get_data(Tag.METADATA_SETTING)
```

**top**

Top coordinate. Writable.

**Returns** int

**topil** (*channel=None, apply\_icc=False*)

Get PIL Image of the layer.

**Parameters**

- **channel** – Which channel to return; e.g., 0 for ‘R’ channel in RGB image. See `ChannelID`. When *None*, the method returns all the channels supported by PIL modes.
- **apply\_icc** – Whether to apply ICC profile conversion to sRGB.

**Returns** `PIL.Image`, or *None* if the layer has no pixels.

Example:

```
from psd_tools.constants import ChannelID

image = layer.topil()
red = layer.topil(ChannelID.CHANNEL_0)
alpha = layer.topil(ChannelID.TRANSPARENCY_MASK)
```

---

**Note:** Not all of the PSD image modes are supported in `PIL.Image`. For example, ‘CMYK’ mode cannot include alpha channel in PIL. In this case, `topil` drops alpha channel.

---

**vector\_mask**

Returns vector mask associated with this layer.

**Returns** *VectorMask* or *None*

**visible**

Layer visibility. Doesn’t take group visibility in account. Writable.

**Returns** *bool*

**width**

Width of the layer.

**Returns** int

### 3.4.2 Group

**class** `psd_tools.api.layers.Group(*args)`

Group of layers.

Example:

```
group = psd[1]
for layer in group:
    if layer.kind == 'pixel':
        print(layer.name)
```

**bbox**

(left, top, right, bottom) tuple.

**blend\_mode**

Blend mode of this layer. Writable.

Example:

```
from psd_tools.constants import BlendMode
if layer.blend_mode == BlendMode.NORMAL:
    layer.blend_mode = BlendMode.SCREEN
```

**Returns** *BlendMode*.

**clip\_layers**

Clip layers associated with this layer.

To compose clipping layers:

```
from psd_tools import compose
clip_mask = compose(layer.clip_layers)
```

**Returns** list of layers

**compose** (*force=False, bbox=None, layer\_filter=None, context=None, color=None*)

Compose layer and masks (mask, vector mask, and clipping layers).

**Returns** PIL Image object, or None if the layer has no pixels.

**composite** (*viewport=None, force=False, color=1.0, alpha=0.0, layer\_filter=None*)

Composite layer and masks (mask, vector mask, and clipping layers).

**Parameters**

- **viewport** – Viewport bounding box specified by (x1, y1, x2, y2) tuple. Default is the layer's bbox.
- **force** – Boolean flag to force vector drawing.
- **color** – Backdrop color specified by scalar or tuple of scalar. The color value should be in [0.0, 1.0]. For example, (1., 0., 0.) specifies red in RGB color mode.
- **alpha** – Backdrop alpha in [0.0, 1.0].

- **layer\_filter** – Callable that takes a layer as argument and returns whether if the layer is composited. Default is `is_visible()`.

**Returns** `PIL.Image`.

**descendants** (*include\_clip=True*)

Return a generator to iterate over all descendant layers.

Example:

```
# Iterate over all layers
for layer in psd.descendants():
    print(layer)

# Iterate over all layers in reverse order
for layer in reversed(list(psd.descendants())):
    print(layer)
```

**Parameters include\_clip** – include clipping layers.

**effects**

Layer effects.

**Returns** `Effects`

**static extract\_bbox** (*layers, include\_invisible=False*)

Returns a bounding box for `layers` or (0, 0, 0, 0) if the layers have no bounding box.

**Parameters include\_invisible** – include invisible layers in calculation.

**Returns** tuple of four int

**has\_clip\_layers** ()

Returns True if the layer has associated clipping.

**Returns** `bool`

**has\_effects** ()

Returns True if the layer has effects.

**Returns** `bool`

**has\_mask** ()

Returns True if the layer has a mask.

**Returns** `bool`

**has\_origination** ()

Returns True if the layer has live shape properties.

**Returns** `bool`

**has\_pixels** ()

Returns True if the layer has associated pixels. When this is True, `topil` method returns `PIL.Image`.

**Returns** `bool`

**has\_stroke** ()

Returns True if the shape has a stroke.

**has\_vector\_mask** ()

Returns True if the layer has a vector mask.

**Returns** `bool`

**height**

Height of the layer.

**Returns** *int*

**is\_group()**

Return True if the layer is a group.

**Returns** *bool*

**is\_visible()**

Layer visibility. Takes group visibility in account.

**Returns** *bool*

**kind**

Kind of this layer, such as group, pixel, shape, type, smartobject, or psdimage. Class name without *layer* suffix.

**Returns** *str*

**layer\_id**

Layer ID.

**Returns** *int* layer id. if the layer is not assigned an id, -1.

**mask**

Returns mask associated with this layer.

**Returns** *Mask* or *None*

**name**

Layer name. Writable.

**Returns** *str*

**numpy** (*channel=None*)

Get NumPy array of the layer.

**Parameters** **channel** – Which channel to return, can be ‘color’, ‘shape’, ‘alpha’, or ‘mask’. Default is ‘color+alpha’.

**Returns** *numpy.ndarray* or *None* if there is no pixel.

**offset**

(left, top) tuple. Writable.

**Returns** *tuple*

**opacity**

Opacity of this layer in [0, 255] range. Writable.

**Returns** *int*

**origination**

Property for a list of live shapes or a line.

Some of the vector masks have associated live shape properties, that are Photoshop feature to handle primitive shapes such as a rectangle, an ellipse, or a line. Vector masks without live shape properties are plain path objects.

See [\*psd\\_tools.api.shape\*](#).

**Returns** List of *Invalidated*, *Rectangle*, *RoundedRectangle*, *Ellipse*, or *Line*.

**parent**

Parent of this layer.

**size**

(width, height) tuple.

**Returns** *tuple*

**stroke**

Property for strokes.

**tagged\_blocks**

Layer tagged blocks that is a dict-like container of settings.

See `psd_tools.constants.Tag` for available keys.

**Returns** *TaggedBlocks* or *None*.

Example:

```
from psd_tools.constants import Tag
metadata = layer.tagged_blocks.get_data(Tag.METADATA_SETTING)
```

**topil** (*channel=None, apply\_icc=False*)

Get PIL Image of the layer.

**Parameters**

- **channel** – Which channel to return; e.g., 0 for ‘R’ channel in RGB image. See `ChannelID`. When *None*, the method returns all the channels supported by PIL modes.
- **apply\_icc** – Whether to apply ICC profile conversion to sRGB.

**Returns** `PIL`. Image, or *None* if the layer has no pixels.

Example:

```
from psd_tools.constants import ChannelID

image = layer.topil()
red = layer.topil(ChannelID.CHANNEL_0)
alpha = layer.topil(ChannelID.TRANSPARENCY_MASK)
```

---

**Note:** Not all of the PSD image modes are supported in `PIL`. Image. For example, ‘CMYK’ mode cannot include alpha channel in `PIL`. In this case, `topil` drops alpha channel.

---

**vector\_mask**

Returns vector mask associated with this layer.

**Returns** *VectorMask* or *None*

**visible**

Layer visibility. Doesn’t take group visibility in account. Writable.

**Returns** *bool*

**width**

Width of the layer.

**Returns** *int*

### 3.4.3 PixelLayer

**class** `psd_tools.api.layers.PixelLayer` (*psd, record, channels, parent*)

Layer that has rasterized image in pixels.

Example:

```
assert layer.kind == 'pixel':
image = layer.topil()
image.save('layer.png')

composed_image = layer.compose()
composed_image.save('composed-layer.png')
```

**bbox**

(left, top, right, bottom) tuple.

**blend\_mode**

Blend mode of this layer. Writable.

Example:

```
from psd_tools.constants import BlendMode
if layer.blend_mode == BlendMode.NORMAL:
    layer.blend_mode = BlendMode.SCREEN
```

**Returns** *BlendMode*.

**bottom**

Bottom coordinate.

**Returns** int

**clip\_layers**

Clip layers associated with this layer.

To compose clipping layers:

```
from psd_tools import compose
clip_mask = compose(layer.clip_layers)
```

**Returns** list of layers

**compose** (*force=False, bbox=None, layer\_filter=None*)

Deprecated, use *composite()*.

Compose layer and masks (mask, vector mask, and clipping layers).

Note that the resulting image size is not necessarily equal to the layer size due to different mask dimensions. The offset of the composed image is stored at *.info['offset']* attribute of *PIL.Image*.

**Parameters** **bbox** – Viewport bounding box specified by (x1, y1, x2, y2) tuple.

**Returns** *PIL.Image*, or *None* if the layer has no pixel.

**composite** (*viewport=None, force=False, color=1.0, alpha=0.0, layer\_filter=None*)

Composite layer and masks (mask, vector mask, and clipping layers).

**Parameters**

- **viewport** – Viewport bounding box specified by (x1, y1, x2, y2) tuple. Default is the layer's bbox.
- **force** – Boolean flag to force vector drawing.
- **color** – Backdrop color specified by scalar or tuple of scalar. The color value should be in [0.0, 1.0]. For example, (1., 0., 0.) specifies red in RGB color mode.
- **alpha** – Backdrop alpha in [0.0, 1.0].
- **layer\_filter** – Callable that takes a layer as argument and returns whether if the layer is composited. Default is *is\_visible()*.

**Returns** `PIL.Image`.

#### **effects**

Layer effects.

**Returns** `Effects`

#### **has\_clip\_layers()**

Returns True if the layer has associated clipping.

**Returns** `bool`

#### **has\_effects()**

Returns True if the layer has effects.

**Returns** `bool`

#### **has\_mask()**

Returns True if the layer has a mask.

**Returns** `bool`

#### **has\_origination()**

Returns True if the layer has live shape properties.

**Returns** `bool`

#### **has\_pixels()**

Returns True if the layer has associated pixels. When this is True, *topil* method returns `PIL.Image`.

**Returns** `bool`

#### **has\_stroke()**

Returns True if the shape has a stroke.

#### **has\_vector\_mask()**

Returns True if the layer has a vector mask.

**Returns** `bool`

#### **height**

Height of the layer.

**Returns** `int`

#### **is\_group()**

Return True if the layer is a group.

**Returns** `bool`

#### **is\_visible()**

Layer visibility. Takes group visibility in account.

**Returns** `bool`

**kind**

Kind of this layer, such as group, pixel, shape, type, smartobject, or psdimage. Class name without *layer* suffix.

**Returns** *str*

**layer\_id**

Layer ID.

**Returns** int layer id. if the layer is not assigned an id, -1.

**left**

Left coordinate. Writable.

**Returns** int

**mask**

Returns mask associated with this layer.

**Returns** *Mask* or *None*

**name**

Layer name. Writable.

**Returns** *str*

**numpy** (*channel=None*)

Get NumPy array of the layer.

**Parameters** **channel** – Which channel to return, can be ‘color’, ‘shape’, ‘alpha’, or ‘mask’. Default is ‘color+alpha’.

**Returns** `numpy.ndarray` or *None* if there is no pixel.

**offset**

(left, top) tuple. Writable.

**Returns** *tuple*

**opacity**

Opacity of this layer in [0, 255] range. Writable.

**Returns** int

**origination**

Property for a list of live shapes or a line.

Some of the vector masks have associated live shape properties, that are Photoshop feature to handle primitive shapes such as a rectangle, an ellipse, or a line. Vector masks without live shape properties are plain path objects.

See `psd_tools.api.shape`.

**Returns** List of *Invalidated*, *Rectangle*, *RoundedRectangle*, *Ellipse*, or *Line*.

**parent**

Parent of this layer.

**right**

Right coordinate.

**Returns** int

**size**

(width, height) tuple.



**Returns** *tuple*

#### **stroke**

Property for strokes.

#### **tagged\_blocks**

Layer tagged blocks that is a dict-like container of settings.

See `psd_tools.constants.Tag` for available keys.

**Returns** *TaggedBlocks* or *None*.

Example:

```
from psd_tools.constants import Tag
metadata = layer.tagged_blocks.get_data(Tag.METADATA_SETTING)
```

#### **top**

Top coordinate. Writable.

**Returns** *int*

#### **topil** (*channel=None, apply\_icc=False*)

Get PIL Image of the layer.

##### **Parameters**

- **channel** – Which channel to return; e.g., 0 for ‘R’ channel in RGB image. See `ChannelID`. When *None*, the method returns all the channels supported by PIL modes.
- **apply\_icc** – Whether to apply ICC profile conversion to sRGB.

**Returns** `PIL`.Image, or *None* if the layer has no pixels.

Example:

```
from psd_tools.constants import ChannelID

image = layer.topil()
red = layer.topil(ChannelID.CHANNEL_0)
alpha = layer.topil(ChannelID.TRANSPARENCY_MASK)
```

---

**Note:** Not all of the PSD image modes are supported in `PIL`.Image. For example, ‘CMYK’ mode cannot include alpha channel in `PIL`. In this case, `topil` drops alpha channel.

---

#### **vector\_mask**

Returns vector mask associated with this layer.

**Returns** *VectorMask* or *None*

#### **visible**

Layer visibility. Doesn’t take group visibility in account. Writable.

**Returns** *bool*

#### **width**

Width of the layer.

**Returns** *int*

### 3.4.4 ShapeLayer

**class** `psd_tools.api.layers.ShapeLayer` (*psd, record, channels, parent*)

Layer that has drawing in vector mask.

**bbox**

(left, top, right, bottom) tuple.

**blend\_mode**

Blend mode of this layer. Writable.

Example:

```
from psd_tools.constants import BlendMode
if layer.blend_mode == BlendMode.NORMAL:
    layer.blend_mode = BlendMode.SCREEN
```

**Returns** *BlendMode*.

**bottom**

Bottom coordinate.

**Returns** *int*

**clip\_layers**

Clip layers associated with this layer.

To compose clipping layers:

```
from psd_tools import compose
clip_mask = compose(layer.clip_layers)
```

**Returns** list of layers

**compose** (*force=False, bbox=None, layer\_filter=None*)

Deprecated, use *composite()*.

Compose layer and masks (mask, vector mask, and clipping layers).

Note that the resulting image size is not necessarily equal to the layer size due to different mask dimensions. The offset of the composed image is stored at *.info['offset']* attribute of *PIL.Image*.

**Parameters** **bbox** – Viewport bounding box specified by (x1, y1, x2, y2) tuple.

**Returns** *PIL.Image*, or *None* if the layer has no pixel.

**composite** (*viewport=None, force=False, color=1.0, alpha=0.0, layer\_filter=None*)

Composite layer and masks (mask, vector mask, and clipping layers).

**Parameters**

- **viewport** – Viewport bounding box specified by (x1, y1, x2, y2) tuple. Default is the layer's bbox.
- **force** – Boolean flag to force vector drawing.
- **color** – Backdrop color specified by scalar or tuple of scalar. The color value should be in [0.0, 1.0]. For example, (1., 0., 0.) specifies red in RGB color mode.
- **alpha** – Backdrop alpha in [0.0, 1.0].

- **layer\_filter** – Callable that takes a layer as argument and returns whether if the layer is composited. Default is `is_visible()`.

**Returns** `PIL.Image`.

#### **effects**

Layer effects.

**Returns** `Effects`

#### **has\_clip\_layers()**

Returns True if the layer has associated clipping.

**Returns** `bool`

#### **has\_effects()**

Returns True if the layer has effects.

**Returns** `bool`

#### **has\_mask()**

Returns True if the layer has a mask.

**Returns** `bool`

#### **has\_origination()**

Returns True if the layer has live shape properties.

**Returns** `bool`

#### **has\_pixels()**

Returns True if the layer has associated pixels. When this is True, `topil` method returns `PIL.Image`.

**Returns** `bool`

#### **has\_stroke()**

Returns True if the shape has a stroke.

#### **has\_vector\_mask()**

Returns True if the layer has a vector mask.

**Returns** `bool`

#### **height**

Height of the layer.

**Returns** `int`

#### **is\_group()**

Return True if the layer is a group.

**Returns** `bool`

#### **is\_visible()**

Layer visibility. Takes group visibility in account.

**Returns** `bool`

#### **kind**

Kind of this layer, such as group, pixel, shape, type, smartobject, or psdimage. Class name without *layer* suffix.

**Returns** `str`

#### **layer\_id**

Layer ID.

**Returns** int layer id. if the layer is not assigned an id, -1.

**left**

Left coordinate. Writable.

**Returns** int

**mask**

Returns mask associated with this layer.

**Returns** *Mask* or *None*

**name**

Layer name. Writable.

**Returns** *str*

**numpy** (*channel=None*)

Get NumPy array of the layer.

**Parameters** **channel** – Which channel to return, can be ‘color’, ‘shape’, ‘alpha’, or ‘mask’.  
Default is ‘color+alpha’.

**Returns** `numpy.ndarray` or `None` if there is no pixel.

**offset**

(left, top) tuple. Writable.

**Returns** *tuple*

**opacity**

Opacity of this layer in [0, 255] range. Writable.

**Returns** int

**origination**

Property for a list of live shapes or a line.

Some of the vector masks have associated live shape properties, that are Photoshop feature to handle primitive shapes such as a rectangle, an ellipse, or a line. Vector masks without live shape properties are plain path objects.

See `psd_tools.api.shape`.

**Returns** List of *Invalidated*, *Rectangle*, *RoundedRectangle*, *Ellipse*, or *Line*.

**parent**

Parent of this layer.

**right**

Right coordinate.

**Returns** int

**size**

(width, height) tuple.

**Returns** *tuple*

**stroke**

Property for strokes.

**tagged\_blocks**

Layer tagged blocks that is a dict-like container of settings.

See `psd_tools.constants.Tag` for available keys.

**Returns** *TaggedBlocks* or *None*.

Example:

```
from psd_tools.constants import Tag
metadata = layer.tagged_blocks.get_data(Tag.METADATA_SETTING)
```

#### **top**

Top coordinate. Writable.

**Returns** int

#### **topil** (*channel=None, apply\_icc=False*)

Get PIL Image of the layer.

##### **Parameters**

- **channel** – Which channel to return; e.g., 0 for ‘R’ channel in RGB image. See *ChannelID*. When *None*, the method returns all the channels supported by PIL modes.
- **apply\_icc** – Whether to apply ICC profile conversion to sRGB.

**Returns** PIL Image, or *None* if the layer has no pixels.

Example:

```
from psd_tools.constants import ChannelID

image = layer.topil()
red = layer.topil(ChannelID.CHANNEL_0)
alpha = layer.topil(ChannelID.TRANSPARENCY_MASK)
```

---

**Note:** Not all of the PSD image modes are supported in PIL Image. For example, ‘CMYK’ mode cannot include alpha channel in PIL. In this case, `topil` drops alpha channel.

---

#### **vector\_mask**

Returns vector mask associated with this layer.

**Returns** *VectorMask* or *None*

#### **visible**

Layer visibility. Doesn’t take group visibility in account. Writable.

**Returns** *bool*

#### **width**

Width of the layer.

**Returns** int

### 3.4.5 SmartObjectLayer

**class** `psd_tools.api.layers.SmartObjectLayer` (*psd, record, channels, parent*)

Layer that inserts external data.

Use *smart\_object* attribute to get the external data. See *SmartObject*.

Example:

```
import io
if layer.smart_object.filetype == 'jpg':
    image = Image.open(io.BytesIO(layer.smart_object.data))
```

**bbox**

(left, top, right, bottom) tuple.

**blend\_mode**

Blend mode of this layer. Writable.

Example:

```
from psd_tools.constants import BlendMode
if layer.blend_mode == BlendMode.NORMAL:
    layer.blend_mode = BlendMode.SCREEN
```

**Returns** *BlendMode*.

**bottom**

Bottom coordinate.

**Returns** int

**clip\_layers**

Clip layers associated with this layer.

To compose clipping layers:

```
from psd_tools import compose
clip_mask = compose(layer.clip_layers)
```

**Returns** list of layers

**compose** (*force=False, bbox=None, layer\_filter=None*)

Deprecated, use *composite()*.

Compose layer and masks (mask, vector mask, and clipping layers).

Note that the resulting image size is not necessarily equal to the layer size due to different mask dimensions. The offset of the composed image is stored at *.info['offset']* attribute of *PIL.Image*.

**Parameters** **bbox** – Viewport bounding box specified by (x1, y1, x2, y2) tuple.

**Returns** *PIL.Image*, or *None* if the layer has no pixel.

**composite** (*viewport=None, force=False, color=1.0, alpha=0.0, layer\_filter=None*)

Composite layer and masks (mask, vector mask, and clipping layers).

**Parameters**

- **viewport** – Viewport bounding box specified by (x1, y1, x2, y2) tuple. Default is the layer's bbox.
- **force** – Boolean flag to force vector drawing.
- **color** – Backdrop color specified by scalar or tuple of scalar. The color value should be in [0.0, 1.0]. For example, (1., 0., 0.) specifies red in RGB color mode.
- **alpha** – Backdrop alpha in [0.0, 1.0].

- **layer\_filter** – Callable that takes a layer as argument and returns whether if the layer is composited. Default is `is_visible()`.

**Returns** `PIL.Image`.

#### **effects**

Layer effects.

**Returns** `Effects`

#### **has\_clip\_layers()**

Returns True if the layer has associated clipping.

**Returns** `bool`

#### **has\_effects()**

Returns True if the layer has effects.

**Returns** `bool`

#### **has\_mask()**

Returns True if the layer has a mask.

**Returns** `bool`

#### **has\_origination()**

Returns True if the layer has live shape properties.

**Returns** `bool`

#### **has\_pixels()**

Returns True if the layer has associated pixels. When this is True, `topil` method returns `PIL.Image`.

**Returns** `bool`

#### **has\_stroke()**

Returns True if the shape has a stroke.

#### **has\_vector\_mask()**

Returns True if the layer has a vector mask.

**Returns** `bool`

#### **height**

Height of the layer.

**Returns** `int`

#### **is\_group()**

Return True if the layer is a group.

**Returns** `bool`

#### **is\_visible()**

Layer visibility. Takes group visibility in account.

**Returns** `bool`

#### **kind**

Kind of this layer, such as group, pixel, shape, type, smartobject, or psdimage. Class name without *layer* suffix.

**Returns** `str`

#### **layer\_id**

Layer ID.

**Returns** int layer id. if the layer is not assigned an id, -1.

**left**

Left coordinate. Writable.

**Returns** int

**mask**

Returns mask associated with this layer.

**Returns** *Mask* or *None*

**name**

Layer name. Writable.

**Returns** *str*

**numpy** (*channel=None*)

Get NumPy array of the layer.

**Parameters** **channel** – Which channel to return, can be ‘color’, ‘shape’, ‘alpha’, or ‘mask’.  
Default is ‘color+alpha’.

**Returns** `numpy.ndarray` or `None` if there is no pixel.

**offset**

(left, top) tuple. Writable.

**Returns** *tuple*

**opacity**

Opacity of this layer in [0, 255] range. Writable.

**Returns** int

**origination**

Property for a list of live shapes or a line.

Some of the vector masks have associated live shape properties, that are Photoshop feature to handle primitive shapes such as a rectangle, an ellipse, or a line. Vector masks without live shape properties are plain path objects.

See `psd_tools.api.shape`.

**Returns** List of *Invalidated*, *Rectangle*, *RoundedRectangle*, *Ellipse*, or *Line*.

**parent**

Parent of this layer.

**right**

Right coordinate.

**Returns** int

**size**

(width, height) tuple.

**Returns** *tuple*

**smart\_object**

Associated smart object.

**Returns** *SmartObject*.

**stroke**

Property for strokes.



**tagged\_blocks**

Layer tagged blocks that is a dict-like container of settings.

See `psd_tools.constants.Tag` for available keys.

**Returns** `TaggedBlocks` or `None`.

Example:

```
from psd_tools.constants import Tag
metadata = layer.tagged_blocks.get_data(Tag.METADATA_SETTING)
```

**top**

Top coordinate. Writable.

**Returns** `int`

**topil** (*channel=None, apply\_icc=False*)

Get PIL Image of the layer.

**Parameters**

- **channel** – Which channel to return; e.g., 0 for ‘R’ channel in RGB image. See `ChannelID`. When `None`, the method returns all the channels supported by PIL modes.
- **apply\_icc** – Whether to apply ICC profile conversion to sRGB.

**Returns** `PIL.Image`, or `None` if the layer has no pixels.

Example:

```
from psd_tools.constants import ChannelID

image = layer.topil()
red = layer.topil(ChannelID.CHANNEL_0)
alpha = layer.topil(ChannelID.TRANSPARENCY_MASK)
```

---

**Note:** Not all of the PSD image modes are supported in `PIL.Image`. For example, ‘CMYK’ mode cannot include alpha channel in PIL. In this case, `topil` drops alpha channel.

---

**vector\_mask**

Returns vector mask associated with this layer.

**Returns** `VectorMask` or `None`

**visible**

Layer visibility. Doesn’t take group visibility in account. Writable.

**Returns** `bool`

**width**

Width of the layer.

**Returns** `int`

### 3.4.6 TypeLayer

**class** `psd_tools.api.layers.TypeLayer` (\*args)

Layer that has text and styling information for fonts or paragraphs.

Text is accessible at `text` property. Styling information for paragraphs is in `engine_dict`. Document styling information such as font list is in `resource_dict`.

Currently, textual information is read-only.

Example:

```
if layer.kind == 'type':
    print(layer.text)
    print(layer.engine_dict['StyleRun'])

    # Extract font for each substring in the text.
    text = layer.engine_dict['Editor']['Text'].value
    fontset = layer.resource_dict['FontSet']
    runlength = layer.engine_dict['StyleRun']['RunLengthArray']
    rundata = layer.engine_dict['StyleRun']['RunArray']
    index = 0
    for length, style in zip(runlength, rundata):
        substring = text[index:index + length]
        stylesheet = style['StyleSheet']['StyleSheetData']
        font = fontset[stylesheet['Font']]
        print('%r gets %s' % (substring, font))
        index += length
```

#### **bbox**

(left, top, right, bottom) tuple.

#### **blend\_mode**

Blend mode of this layer. Writable.

Example:

```
from psd_tools.constants import BlendMode
if layer.blend_mode == BlendMode.NORMAL:
    layer.blend_mode = BlendMode.SCREEN
```

**Returns** *BlendMode*.

#### **bottom**

Bottom coordinate.

**Returns** int

#### **clip\_layers**

Clip layers associated with this layer.

To compose clipping layers:

```
from psd_tools import compose
clip_mask = compose(layer.clip_layers)
```

**Returns** list of layers

#### **compose** (*force=False, bbox=None, layer\_filter=None*)

Deprecated, use `composite()`.

Compose layer and masks (mask, vector mask, and clipping layers).

Note that the resulting image size is not necessarily equal to the layer size due to different mask dimensions. The offset of the composed image is stored at `.info['offset']` attribute of `PIL.Image`.

**Parameters** **bbox** – Viewport bounding box specified by (x1, y1, x2, y2) tuple.

**Returns** `PIL.Image`, or `None` if the layer has no pixel.

**composite** (*viewport=None, force=False, color=1.0, alpha=0.0, layer\_filter=None*)

Composite layer and masks (mask, vector mask, and clipping layers).

**Parameters**

- **viewport** – Viewport bounding box specified by (x1, y1, x2, y2) tuple. Default is the layer's bbox.
- **force** – Boolean flag to force vector drawing.
- **color** – Backdrop color specified by scalar or tuple of scalar. The color value should be in [0.0, 1.0]. For example, (1., 0., 0.) specifies red in RGB color mode.
- **alpha** – Backdrop alpha in [0.0, 1.0].
- **layer\_filter** – Callable that takes a layer as argument and returns whether if the layer is composited. Default is `is_visible()`.

**Returns** `PIL.Image`.

**document\_resources**

Resource set relevant to the document.

**effects**

Layer effects.

**Returns** `Effects`

**engine\_dict**

Styling information dict.

**has\_clip\_layers** ()

Returns True if the layer has associated clipping.

**Returns** `bool`

**has\_effects** ()

Returns True if the layer has effects.

**Returns** `bool`

**has\_mask** ()

Returns True if the layer has a mask.

**Returns** `bool`

**has\_origination** ()

Returns True if the layer has live shape properties.

**Returns** `bool`

**has\_pixels** ()

Returns True if the layer has associated pixels. When this is True, `topil` method returns `PIL.Image`.

**Returns** `bool`

**has\_stroke** ()

Returns True if the shape has a stroke.

**has\_vector\_mask** ()

Returns True if the layer has a vector mask.

**Returns** `bool`

**height**

Height of the layer.

**Returns** *int*

**is\_group()**

Return True if the layer is a group.

**Returns** *bool*

**is\_visible()**

Layer visibility. Takes group visibility in account.

**Returns** *bool*

**kind**

Kind of this layer, such as group, pixel, shape, type, smartobject, or psdimage. Class name without *layer* suffix.

**Returns** *str*

**layer\_id**

Layer ID.

**Returns** *int* layer id. if the layer is not assigned an id, -1.

**left**

Left coordinate. Writable.

**Returns** *int*

**mask**

Returns mask associated with this layer.

**Returns** *Mask* or *None*

**name**

Layer name. Writable.

**Returns** *str*

**numpy (channel=None)**

Get NumPy array of the layer.

**Parameters** **channel** – Which channel to return, can be ‘color’, ‘shape’, ‘alpha’, or ‘mask’. Default is ‘color+alpha’.

**Returns** *numpy.ndarray* or *None* if there is no pixel.

**offset**

(left, top) tuple. Writable.

**Returns** *tuple*

**opacity**

Opacity of this layer in [0, 255] range. Writable.

**Returns** *int*

**origination**

Property for a list of live shapes or a line.

Some of the vector masks have associated live shape properties, that are Photoshop feature to handle primitive shapes such as a rectangle, an ellipse, or a line. Vector masks without live shape properties are plain path objects.

See `psd_tools.api.shape`.

**Returns** List of *Invalidated*, *Rectangle*, *RoundedRectangle*, *Ellipse*, or *Line*.

#### parent

Parent of this layer.

#### resource\_dict

Resource set.

#### right

Right coordinate.

**Returns** int

#### size

(width, height) tuple.

**Returns** tuple

#### stroke

Property for strokes.

#### tagged\_blocks

Layer tagged blocks that is a dict-like container of settings.

See `psd_tools.constants.Tag` for available keys.

**Returns** *TaggedBlocks* or *None*.

Example:

```
from psd_tools.constants import Tag
metadata = layer.tagged_blocks.get_data(Tag.METADATA_SETTING)
```

#### text

Text in the layer. Read-only.

---

**Note:** New-line character in Photoshop is ‘\r’.

---

#### top

Top coordinate. Writable.

**Returns** int

#### topil (*channel=None*, *apply\_icc=False*)

Get PIL Image of the layer.

##### Parameters

- **channel** – Which channel to return; e.g., 0 for ‘R’ channel in RGB image. See *ChannelID*. When *None*, the method returns all the channels supported by PIL modes.
- **apply\_icc** – Whether to apply ICC profile conversion to sRGB.

**Returns** PIL Image, or *None* if the layer has no pixels.

Example:

```
from psd_tools.constants import ChannelID
image = layer.topil()
```

(continues on next page)

(continued from previous page)

```
red = layer.topil(ChannelID.CHANNEL_0)
alpha = layer.topil(ChannelID.TRANSPARENCY_MASK)
```

---

**Note:** Not all of the PSD image modes are supported in `PIL.Image`. For example, ‘CMYK’ mode cannot include alpha channel in `PIL`. In this case, `topil` drops alpha channel.

---

**transform**

Matrix  $(xx, xy, yx, yy, tx, ty)$  applies affine transformation.

**vector\_mask**

Returns vector mask associated with this layer.

**Returns** *VectorMask* or *None*

**visible**

Layer visibility. Doesn’t take group visibility in account. Writable.

**Returns** *bool*

**warp**

Warp configuration.

**width**

Width of the layer.

**Returns** *int*

## 3.5 psd\_tools.api.mask

Mask module.

### 3.5.1 Mask

**class** `psd_tools.api.mask.Mask` (*layer*)

Mask data attached to a layer.

There are two distinct internal mask data: user mask and vector mask. User mask refers any pixel-based mask whereas vector mask refers a mask from a shape path. Internally, two masks are combined and referred real mask.

**background\_color**

Background color.

**bbox**

BBox

**bottom**

Bottom coordinate.

**disabled**

Disabled.

**flags**

Flags.

**height**

Height.

**left**

Left coordinate.

**parameters**

Parameters.

**real\_flags**

Real flag.

**right**

Right coordinate.

**size**

(Width, Height) tuple.

**top**

Top coordinate.

**topil** (*real=True, \*\*kwargs*)

Get PIL Image of the mask.

**Parameters** **real** – When True, returns pixel + vector mask combined.**Returns** PIL Image object, or None if the mask is empty.**width**

Width.

## 3.6 psd\_tools.api.shape

Shape module.

In PSD/PSB, shapes are all represented as *VectorMask* in each layer, and optionally there might be *Origination* object to control live shape properties and *Stroke* to specify how outline is stylized.

### 3.6.1 VectorMask

**class** `psd_tools.api.shape.VectorMask` (*data*)

Vector mask data.

Vector mask is a resolution-independent mask that consists of one or more Path objects. In Photoshop, all the path objects are represented as Bezier curves. Check *paths* property for how to deal with path objects.

**bbox**

Bounding box tuple (left, top, right, bottom) in relative coordinates, where top-left corner is (0., 0.) and bottom-right corner is (1., 1.).

**Returns** *tuple***clipboard\_record**

Clipboard record containing bounding box information.

Depending on the Photoshop version, this field can be *None*.

**disabled**

If the mask is disabled.

**initial\_fill\_rule**

Initial fill rule.

When 0, fill inside of the path. When 1, fill outside of the shape.

**Returns** *int*

**inverted**

Invert the mask.

**not\_linked**

If the knots are not linked.

**paths**

List of *Subpath*. Subpath is a list-like structure that contains one or more *Knot* items. Knot contains relative coordinates of control points for a Bezier curve. *index* indicates which origination item the subpath belongs, and *operation* indicates how to combine multiple shape paths.

In PSD, path fill rule is even-odd.

Example:

```
for subpath in layer.vector_mask.paths:
    anchors = [(
        int(knot.anchor[1] * psd.width),
        int(knot.anchor[0] * psd.height),
    ) for knot in subpath]
```

**Returns** List of Subpath.

## 3.6.2 Stroke

**class** `psd_tools.api.shape.Stroke` (*data*)

Stroke contains decorative information for strokes.

This is a thin wrapper around *Descriptor* structure. Check *\_data* attribute to get the raw data.

**blend\_mode**

Blend mode.

**content**

Fill effect.

**enabled**

If the stroke is enabled.

**fill\_enabled**

If the stroke fill is enabled.

**line\_alignment**

Alignment, one of *inner*, *outer*, *center*.

**line\_cap\_type**

Cap type, one of *butt*, *round*, *square*.

**line\_dash\_offset**

Line dash offset in float.

**Returns** float

**line\_dash\_set**

Line dash set in list of `UnitFloat`.



**Returns** list

**line\_join\_type**

Join type, one of *miter*, *round*, *bevel*.

**line\_width**

Stroke width in float.

**miter\_limit**

Miter limit in float.

**opacity**

Opacity value.

**stroke\_adjust**

Stroke adjust

### 3.6.3 Origination

Origination keeps live shape properties for some of the primitive shapes. Origination objects are accessible via origination property of layers. Following primitive shapes are defined: *Invalidated*, *Line*, *Rectangle*, *Ellipse*, and *RoundedRectangle*.

#### Invalidated

**class** `psd_tools.api.shape.Invalidated` (*data*)

Invalidated live shape.

This equals to a primitive shape that does not provide Live shape properties. Use *VectorMask* to access shape information instead of this origination object.

**invalidated**

**Returns** *bool*

#### Line

**class** `psd_tools.api.shape.Line` (*data*)

Line live shape.

**arrow\_conc**

**Returns** *int*

**arrow\_end**

Line arrow end.

**Returns** *bool*

**arrow\_length**

Line arrow length.

**Returns** *float*

**arrow\_start**

Line arrow start.

**Returns** *bool*

**arrow\_width**

Line arrow width.

**Returns** *float*

**bbox**

Bounding box of the live shape.

**Returns** *Descriptor*

**index**

Origination item index.

**Returns** *int*

**invalidated**

**Returns** *bool*

**line\_end**

Line end.

**Returns** *Descriptor*

**line\_start**

Line start.

**Returns** *Descriptor*

**line\_weight**

Line weight

**Returns** *float*

**origin\_type**

Type of the vector shape.

- 1: *Rectangle*
- 2: *RoundedRectangle*
- 4: *Line*
- 5: *Ellipse*

**Returns** *int*

**resolution**

Resolution.

**Returns** *float*

## Ellipse

**class** `psd_tools.api.shape.Ellipse` (*data*)

Ellipse live shape.

**bbox**

Bounding box of the live shape.

**Returns** *Descriptor*

**index**

Origination item index.

**Returns** *int*

**invalidated**

**Returns** *bool*

**origin\_type**

Type of the vector shape.

- 1: *Rectangle*
- 2: *RoundedRectangle*
- 4: *Line*
- 5: *Ellipse*

**Returns** *int*

**resolution**

Resolution.

**Returns** *float*

## Rectangle

**class** `psd_tools.api.shape.Rectangle` (*data*)

Rectangle live shape.

**bbox**

Bounding box of the live shape.

**Returns** *Descriptor*

**index**

Origination item index.

**Returns** *int*

**invalidated**

**Returns** *bool*

**origin\_type**

Type of the vector shape.

- 1: *Rectangle*
- 2: *RoundedRectangle*
- 4: *Line*
- 5: *Ellipse*

**Returns** *int*

**resolution**

Resolution.

**Returns** *float*

## RoundedRectangle

**class** `psd_tools.api.shape.RoundedRectangle` (*data*)

Rounded rectangle live shape.

**bbox**

Bounding box of the live shape.

**Returns** *Descriptor*

**index**

Origination item index.

**Returns** *int*

**invalidated**

**Returns** *bool*

**origin\_type**

Type of the vector shape.

- 1: *Rectangle*
- 2: *RoundedRectangle*
- 4: *Line*
- 5: *Ellipse*

**Returns** *int*

**radii**

Corner radii of rounded rectangles. The order is top-left, top-right, bottom-left, bottom-right.

**Returns** *Descriptor*

**resolution**

Resolution.

**Returns** *float*

## 3.7 psd\_tools.api.smart\_object

Smart object module.

### 3.7.1 SmartObject

**class** `psd_tools.api.smart_object.SmartObject` (*layer*)

Smart object that represents embedded or external file.

Smart objects are attached to *SmartObjectLayer*.

**data**

Embedded file content, or empty if kind is *external* or *alias*

**filename**

Original file name of the object.

**filesize**

File size of the object.

**filetype**

Preferred file extension, such as *jpg*.

**is\_psd()**

Return True if the file is embedded PSD/PSB.

**kind**

Kind of the link, 'data', 'alias', or 'external'.

**open** (*external\_dir=None*)

Open the smart object as binary IO.

**Parameters** **external\_dir** – Path to the directory of the external file.

Example:

```
with layer.smart_object.open() as f:
    data = f.read()
```

**resolution**

Resolution of the object.

**save** (*filename=None*)

Save the smart object to a file.

**Parameters** **filename** – File name to export. If None, use the embedded name.

**unique\_id**

UUID of the object.

**warp**

Warp parameters.

## 3.8 psd\_tools.constants

Various constants for psd\_tools

### 3.8.1 BlendMode

**class** psd\_tools.constants.BlendMode

Blend modes.

**COLOR** = b'colr'

**COLOR\_BURN** = b'idiv'

**COLOR\_DODGE** = b'div '

**DARKEN** = b'dark'

**DARKER\_COLOR** = b'dkCl'

**DIFFERENCE** = b'diff'

**DISSOLVE** = b'diss'

**DIVIDE** = b'fdiv'

```
EXCLUSION = b'smud'  
HARD_LIGHT = b'hLit'  
HARD_MIX = b'hMix'  
HUE = b'hue '  
LIGHTEN = b'lite'  
LIGHTER_COLOR = b'lgCl'  
LINEAR_BURN = b'lbrn'  
LINEAR_DODGE = b'lddg'  
LINEAR_LIGHT = b'lLit'  
LUMINOSITY = b'lum '  
MULTIPLY = b'mul '  
NORMAL = b'norm'  
OVERLAY = b'over'  
PASS_THROUGH = b'pass'  
PIN_LIGHT = b'pLit'  
SATURATION = b'sat '  
SCREEN = b'scrn'  
SOFT_LIGHT = b'sLit'  
SUBTRACT = b'fsub'  
VIVID_LIGHT = b'vLit'
```

### 3.8.2 ChannelID

```
class psd_tools.constants.ChannelID  
    Channel types.
```

```
CHANNEL_0 = 0  
CHANNEL_1 = 1  
CHANNEL_2 = 2  
CHANNEL_3 = 3  
CHANNEL_4 = 4  
CHANNEL_5 = 5  
CHANNEL_6 = 6  
CHANNEL_7 = 7  
CHANNEL_8 = 8  
CHANNEL_9 = 9  
REAL_USER_LAYER_MASK = -3  
TRANSPARENCY_MASK = -1
```

```
USER_LAYER_MASK = -2
```

### 3.8.3 Clipping

```
class psd_tools.constants.Clipping
    Clipping.
    BASE = 0
    NON_BASE = 1
```

### 3.8.4 ColorMode

```
class psd_tools.constants.ColorMode
    Color mode.
    BITMAP = 0
    CMYK = 4
    DUOTONE = 8
    GRAYSCALE = 1
    INDEXED = 2
    LAB = 9
    MULTICHANNEL = 7
    RGB = 3
    channels = <function ColorMode.channels>
```

### 3.8.5 ColorSpaceID

```
class psd_tools.constants.ColorSpaceID
    Color space types.
    CMYK = 2
    GRAYSCALE = 8
    HSB = 1
    LAB = 7
    RGB = 0
```

### 3.8.6 Compression

```
class psd_tools.constants.Compression
    Compression modes.
    Compression. 0 = Raw Data, 1 = RLE compressed, 2 = ZIP without prediction, 3 = ZIP with prediction.
    RAW = 0
    RLE = 1
```

```
ZIP = 2
ZIP_WITH_PREDICTION = 3
```

### 3.8.7 EffectOSType

```
class psd_tools.constants.EffectOSType
    OS Type keys for Layer Effects.
    BEVEL = b'bevl'
    COMMON_STATE = b'cmnS'
    DROP_SHADOW = b'dsdw'
    INNER_GLOW = b'iglw'
    INNER_SHADOW = b'isdw'
    OUTER_GLOW = b'oglw'
    SOLID_FILL = b'sofi'
```

### 3.8.8 GlobalLayerMaskKind

```
class psd_tools.constants.GlobalLayerMaskKind
    Global layer mask kind.
    COLOR_PROTECTED = 1
    COLOR_SELECTED = 0
    PER_LAYER = 128
```

### 3.8.9 LinkedLayerType

```
class psd_tools.constants.LinkedLayerType
    Linked layer types.
    ALIAS = b'liFA'
    DATA = b'liFD'
    EXTERNAL = b'liFE'
```

### 3.8.10 PathResourceID

```
class psd_tools.constants.PathResourceID
    An enumeration.
    CLIPBOARD = 7
    CLOSED_KNOT_LINKED = 1
    CLOSED_KNOT_UNLINKED = 2
    CLOSED_LENGTH = 0
    INITIAL_FILL = 8
```



```

OPEN_KNOT_LINKED = 4
OPEN_KNOT_UNLINKED = 5
OPEN_LENGTH = 3
PATH_FILL = 6

```

### 3.8.11 PlacedLayerType

```
class psd_tools.constants.PlacedLayerType
```

An enumeration.

```

IMAGE_STACK = 3
RASTER = 2
UNKNOWN = 0
VECTOR = 1

```

### 3.8.12 PrintScaleStyle

```
class psd_tools.constants.PrintScaleStyle
```

Print scale style.

```

CENTERED = 0
SIZE_TO_FIT = 1
USER_DEFINED = 2

```

### 3.8.13 Resource

```
class psd_tools.constants.Resource
```

Image resource keys.

Note the following is not defined for performance reasons.

- `PATH_INFO_10` to `PATH_INFO_989` corresponding to 2010 - 2989
- `PLUGIN_RESOURCES_10` to `PLUGIN_RESOURCES_989` corresponding to 4010 - 4989

```

ALPHA_IDENTIFIERS = 1053
ALPHA_NAMES_PASCAL = 1006
ALPHA_NAMES_UNICODE = 1045
ALTERNATE_DUOTONE_COLORS = 1066
ALTERNATE_SPOT_COLORS = 1067
AUTO_SAVE_FILE_PATH = 1086
AUTO_SAVE_FORMAT = 1087
BACKGROUND_COLOR = 1010
BORDER_INFO = 1009
CAPTION_DIGEST = 1061

```

CAPTION\_PASCAL = 1008  
CLIPPING\_PATH\_NAME = 2999  
COLOR\_HALFTONING\_INFO = 1013  
COLOR\_SAMPLERS\_RESOURCE = 1073  
COLOR\_SAMPLERS\_RESOURCE\_OBSOLETE = 1038  
COLOR\_TRANSFER\_FUNCTION = 1016  
COPYRIGHT\_FLAG = 1034  
COUNT\_INFO = 1080  
DISPLAY\_INFO = 1077  
DISPLAY\_INFO\_OBSOLETE = 1007  
DUOTONE\_HALFTONING\_INFO = 1014  
DUOTONE\_IMAGE\_INFO = 1018  
DUOTONE\_TRANSFER\_FUNCTION = 1017  
EFFECTIVE\_BW = 1019  
EFFECTS\_VISIBLE = 1042  
EPS\_OPTIONS = 1021  
EXIF\_DATA\_1 = 1058  
EXIF\_DATA\_3 = 1059  
GLOBAL\_ALTITUDE = 1049  
GLOBAL\_ANGLE = 1037  
GRAYSCALE\_HALFTONING\_INFO = 1012  
GRAYSCALE\_TRANSFER\_FUNCTION = 1015  
GRID\_AND\_GUIDES\_INFO = 1032  
HDR\_TONING\_INFO = 1070  
ICC\_PROFILE = 1039  
ICC\_UNTAGGED\_PROFILE = 1041  
IDS\_SEED\_NUMBER = 1044  
IMAGE\_MODE\_RAW = 1029  
IMAGE\_READY\_7\_ROLLOVER\_EXPANDED\_STATE = 7003  
IMAGE\_READY\_DATA\_SETS = 7001  
IMAGE\_READY\_DEFAULT\_SELECTED\_STATE = 7002  
IMAGE\_READY\_ROLLOVER\_EXPANDED\_STATE = 7004  
IMAGE\_READY\_SAVE\_LAYER\_SETTINGS = 7005  
IMAGE\_READY\_VARIABLES = 7000  
IMAGE\_READY\_VERSION = 7006  
INDEXED\_COLOR\_TABLE\_COUNT = 1046

```
IPTC_NAA = 1028
JPEG_QUALITY = 1030
JUMP_TO_XPEP = 1052
LAYER_COMPS = 1065
LAYER_GROUPS_ENABLED_ID = 1072
LAYER_GROUP_INFO = 1026
LAYER_SELECTION_IDS = 1069
LAYER_STATE_INFO = 1024
LIGHTROOM_WORKFLOW = 8000
MAC_NSPRINTINFO = 1084
MAC_PAGE_FORMAT_INFO = 1002
MAC_PRINT_MANAGER_INFO = 1001
MEASUREMENT_SCALE = 1074
OBSOLETE1 = 1000
OBSOLETE2 = 1003
OBSOLETE3 = 1020
OBSOLETE4 = 1023
OBSOLETE5 = 1027
ONION_SKINS = 1078
ORIGIN_PATH_INFO = 3000
PATH_INFO_0 = 2000
PATH_INFO_1 = 2001
PATH_INFO_2 = 2002
PATH_INFO_3 = 2003
PATH_INFO_4 = 2004
PATH_INFO_5 = 2005
PATH_INFO_6 = 2006
PATH_INFO_7 = 2007
PATH_INFO_8 = 2008
PATH_INFO_9 = 2009
PATH_INFO_990 = 2990
PATH_INFO_991 = 2991
PATH_INFO_992 = 2992
PATH_INFO_993 = 2993
PATH_INFO_994 = 2994
PATH_INFO_995 = 2995
```

PATH\_INFO\_996 = 2996  
PATH\_INFO\_997 = 2997  
PATH\_SELECTION\_STATE = 1088  
PIXEL\_ASPECT\_RATIO = 1064  
PLUGIN\_RESOURCE\_0 = 4000  
PLUGIN\_RESOURCE\_1 = 4001  
PLUGIN\_RESOURCE\_2 = 4002  
PLUGIN\_RESOURCE\_3 = 4003  
PLUGIN\_RESOURCE\_4 = 4004  
PLUGIN\_RESOURCE\_4990 = 4990  
PLUGIN\_RESOURCE\_4991 = 4991  
PLUGIN\_RESOURCE\_4992 = 4992  
PLUGIN\_RESOURCE\_4993 = 4993  
PLUGIN\_RESOURCE\_4994 = 4994  
PLUGIN\_RESOURCE\_4995 = 4995  
PLUGIN\_RESOURCE\_4996 = 4996  
PLUGIN\_RESOURCE\_4997 = 4997  
PLUGIN\_RESOURCE\_4998 = 4998  
PLUGIN\_RESOURCE\_4999 = 4990  
PLUGIN\_RESOURCE\_5 = 4005  
PLUGIN\_RESOURCE\_6 = 4006  
PLUGIN\_RESOURCE\_7 = 4007  
PLUGIN\_RESOURCE\_8 = 4008  
PLUGIN\_RESOURCE\_9 = 4009  
PRINT\_FLAGS = 1011  
PRINT\_FLAGS\_INFO = 10000  
PRINT\_INFO\_CS2 = 1071  
PRINT\_INFO\_CS5 = 1082  
PRINT\_SCALE = 1062  
PRINT\_STYLE = 1083  
QUICK\_MASK\_INFO = 1022  
RESOLUTION\_INFO = 1005  
SHEET\_DISCLOSURE = 1076  
SLICES = 1050  
SPOT\_HALFTONE = 1043  
THUMBNAIL\_RESOURCE = 1036

```

THUMBNAIL_RESOURCE_PS4 = 1033
TIMELINE_INFO = 1075
TRANSPARENCY_INDEX = 1047
URL = 1035
URL_LIST = 1054
VERSION_INFO = 1057
WATERMARK = 1040
WINDOWS_DEVMODE = 1085
WORKFLOW_URL = 1051
WORKING_PATH = 1025
XMP_METADATA = 1060
is_path_info = <function Resource.is_path_info>
is_plugin_resource = <function Resource.is_plugin_resource>

```

### 3.8.14 SectionDivider

```
class psd_tools.constants.SectionDivider
```

An enumeration.

```

BOUNDING_SECTION_DIVIDER = 3
CLOSED_FOLDER = 2
OPEN_FOLDER = 1
OTHER = 0

```

### 3.8.15 Tag

```
class psd_tools.constants.Tag
```

Tagged blocks keys.

```

ALPHA = b'Alph'
ANIMATION_EFFECTS = b'anFX'
ANNOTATIONS = b'Anno'
ARTBOARD_DATA1 = b'artb'
ARTBOARD_DATA2 = b'artd'
ARTBOARD_DATA3 = b'abdd'
BLACK_AND_WHITE = b'blwh'
BLEND_CLIPPING_ELEMENTS = b'clbl'
BLEND_FILL_OPACITY = b'iOpa'
BLEND_INTERIOR_ELEMENTS = b'infx'
BRIGHTNESS_AND_CONTRAST = b'brit'

```

```
CHANNEL_BLENDING_RESTRICTIONS_SETTING = b'brst'  
CHANNEL_MIXER = b'mixr'  
COLOR_BALANCE = b'blnc'  
COLOR_LOOKUP = b'clrL'  
COMPOSITOR_INFO = b'cinf'  
CONTENT_GENERATOR_EXTRA_DATA = b'CgEd'  
CURVES = b'curv'  
EFFECTS_LAYER = b'lrFX'  
EXPORT_SETTING1 = b'extd'  
EXPORT_SETTING2 = b'extn'  
EXPOSURE = b'expA'  
FILTER_EFFECTS1 = b'FXid'  
FILTER_EFFECTS2 = b'FEid'  
FILTER_EFFECTS3 = b'FELS'  
FILTER_MASK = b'FMsk'  
FOREIGN_EFFECT_ID = b'ffxi'  
FRAMED_GROUP = b'frgb'  
GRADIENT_FILL_SETTING = b'GdFl'  
GRADIENT_MAP = b'grdm'  
HUE_SATURATION = b'hue2'  
HUE_SATURATION_V4 = b'hue '  
INVERT = b'nvrt'  
KNOCKOUT_SETTING = b'knko'  
LAYER = b'Layr'  
LAYER_16 = b'Lr16'  
LAYER_32 = b'Lr32'  
LAYER_ID = b'lyid'  
LAYER_MASK_AS_GLOBAL_MASK = b'lmgm'  
LAYER_NAME_SOURCE_SETTING = b'lnsr'  
LAYER_VERSION = b'lyvr'  
LEVELS = b'levl'  
LINKED_LAYER1 = b'lnkD'  
LINKED_LAYER2 = b'lnk2'  
LINKED_LAYER3 = b'lnk3'  
LINKED_LAYER_EXTERNAL = b'lnkE'  
METADATA_SETTING = b'shmd'
```

```
NESTED_SECTION_DIVIDER_SETTING = b'lsdk'  
OBJECT_BASED_EFFECTS_LAYER_INFO = b'lfx2'  
OBJECT_BASED_EFFECTS_LAYER_INFO_V0 = b'lmfx'  
OBJECT_BASED_EFFECTS_LAYER_INFO_V1 = b'lfxs'  
PATT = b'patt'  
PATTERNS1 = b'Patt'  
PATTERNS2 = b'Pat2'  
PATTERNS3 = b'Pat3'  
PATTERN_DATA = b'shpa'  
PATTERN_FILL_SETTING = b'PtF1'  
PHOTO_FILTER = b'phf1'  
PIXEL_SOURCE_DATA1 = b'PxSc'  
PIXEL_SOURCE_DATA2 = b'PxSD'  
PLACED_LAYER1 = b'plLd'  
PLACED_LAYER2 = b'PlLd'  
POSTERIZE = b'post'  
PROTECTED_SETTING = b'lspf'  
REFERENCE_POINT = b'fxrp'  
SAVING_MERGED_TRANSPARENCY = b'Mtrn'  
SAVING_MERGED_TRANSPARENCY16 = b'Mt16'  
SAVING_MERGED_TRANSPARENCY32 = b'Mt32'  
SECTION_DIVIDER_SETTING = b'lsct'  
SELECTIVE_COLOR = b'selc'  
SHEET_COLOR_SETTING = b'lclr'  
SMART_OBJECT_LAYER_DATA1 = b'SoLd'  
SMART_OBJECT_LAYER_DATA2 = b'SoLE'  
SOLID_COLOR_SHEET_SETTING = b'SoCo'  
TEXT_ENGINE_DATA = b'Txt2'  
THRESHOLD = b'thrs'  
TRANSPARENCY_SHAPES_LAYER = b'tsly'  
TYPE_TOOL_INFO = b'tySh'  
TYPE_TOOL_OBJECT_SETTING = b'TySh'  
UNICODE_LAYER_NAME = b'luni'  
UNICODE_PATH_NAME = b'pths'  
USER_MASK = b'LMsk'  
USING_ALIGNED_RENDERING = b'sn2P'
```

```
VECTOR_MASK_AS_GLOBAL_MASK = b'vmgm'  
VECTOR_MASK_SETTING1 = b'vmsk'  
VECTOR_MASK_SETTING2 = b'vsms'  
VECTOR_ORIGINATION_DATA = b'vogk'  
VECTOR_STROKE_CONTENT_DATA = b'vscg'  
VECTOR_STROKE_DATA = b'vstk'  
VIBRANCE = b'viba'
```

## 3.9 psd\_tools.psd

Low-level API that translates binary data to Python structure.

All the data structure in this subpackage inherits from one of the object defined in `psd_tools.psd.base` module.

### 3.9.1 PSD

```
class psd_tools.psd.PSD(header=NOTHING, color_mode_data=NOTHING, im-  
age_resources=NOTHING, layer_and_mask_information=NOTHING,  
image_data=NOTHING)
```

Low-level PSD file structure that resembles the [specification](#).

Example:

```
from psd_tools.psd import PSD  
  
with open(input_file, 'rb') as f:  
    psd = PSD.read(f)  
  
with open(output_file, 'wb') as f:  
    psd.write(f)
```

#### **header**

See [FileHeader](#).

#### **color\_mode\_data**

See [ColorModeData](#).

#### **image\_resources**

See [ImageResources](#).

#### **layer\_and\_mask\_information**

See [LayerAndMaskInformation](#).

#### **image\_data**

See [ImageData](#).

## 3.10 psd\_tools.psd.base

Base data structures intended for inheritance.



All the data objects in this subpackage inherit from the base classes here. That means, all the data structures in the `psd_tools.psd` subpackage implements the methods of `BaseElement` for serialization and decoding.

Objects that inherit from the `BaseElement` typically gets `attrs` decoration to have data fields.

### 3.10.1 BaseElement

**class** `psd_tools.psd.base.BaseElement`

Base element of various PSD file structs. All the data objects in `psd_tools.psd` subpackage inherit from this class.

**classmethod** `read(cls, fp)`

Read the element from a file-like object.

**write** (*self*, *fp*)

Write the element to a file-like object.

**classmethod** `frombytes(self, data, *args, **kwargs)`

Read the element from bytes.

**tobytes** (*self*, \*args, \*\*kwargs)

Write the element to bytes.

**validate** (*self*)

Validate the attribute.

### 3.10.2 EmptyElement

**class** `psd_tools.psd.base.EmptyElement`

Empty element that does not have a value.

### 3.10.3 ValueElement

**class** `psd_tools.psd.base.ValueElement` (*value=None*)

Single value wrapper that has a *value* attribute.

Pretty printing shows the internal value by default. Inherit with `@attr.s(repr=False)` decorator to keep this behavior.

**value**

Internal value.

### 3.10.4 NumericElement

**class** `psd_tools.psd.base.NumericElement` (*value=0.0*)

Single value element that has a numeric *value* attribute.

### 3.10.5 IntegerElement

**class** `psd_tools.psd.base.IntegerElement` (*value=0*)

Single integer value element that has a *value* attribute.

Use with `@attr.s(repr=False)` decorator.

### 3.10.6 ShortIntegerElement

**class** `psd_tools.psd.base.ShortIntegerElement` (*value=0*)  
Single short integer element that has a *value* attribute.

Use with `@attr.s(repr=False)` decorator.

### 3.10.7 ByteElement

**class** `psd_tools.psd.base.ByteElement` (*value=0*)  
Single 1-byte integer element that has a *value* attribute.

Use with `@attr.s(repr=False)` decorator.

### 3.10.8 BooleanElement

**class** `psd_tools.psd.base.BooleanElement` (*value=False*)  
Single bool value element that has a *value* attribute.

Use with `@attr.s(repr=False)` decorator.

### 3.10.9 StringElement

**class** `psd_tools.psd.base.StringElement` (*value: str = ""*)  
Single unicode string.

**value**  
*str* value

### 3.10.10 ListElement

**class** `psd_tools.psd.base.ListElement` (*items=NOTHING*)  
List-like element that has *items* list.

### 3.10.11 DictElement

**class** `psd_tools.psd.base.DictElement` (*items=NOTHING*)  
Dict-like element that has *items* `OrderedDict`.

## 3.11 psd\_tools.psd.color\_mode\_data

Color mode data structure.

### 3.11.1 ColorModeData

**class** `psd_tools.psd.color_mode_data.ColorModeData` (*value: bytes = b""*)  
Color mode data section of the PSD file.

For indexed color images the data is the color table for the image in a non-interleaved order.

Duotone images also have this data, but the data format is undocumented.

**interleave()**

Returns interleaved color table in bytes.

## 3.12 psd\_tools.psd.descriptor

Descriptor data structure.

Descriptors are basic data structure used throughout PSD files. Descriptor is one kind of serialization protocol for data objects, and enum classes in `psd_tools.terminology` or bytes indicates what kind of descriptor it is.

The class ID can be pre-defined enum if the tag is 4-byte length or plain bytes if the length is arbitrary. They depend on the internal version of Adobe Photoshop but the detail is unknown.

Pretty printing is the best approach to check the descriptor content:

```
from IPython.pretty import pprint
pprint(descriptor)
```

### 3.12.1 Alias

**class** `psd_tools.psd.descriptor.Alias` (*value: bytes = b'x00x00x00x00'*)

Alias structure equivalent to *RawData*.

### 3.12.2 Bool

**class** `psd_tools.psd.descriptor.Bool` (*value=False*)

Bool structure.

**value**

*bool* value

### 3.12.3 Class

**class** `psd_tools.psd.descriptor.Class` (*name: str = "", classID: bytes = b'x00x00x00x00'*)

Class structure.

**name**

*str* value

**classID**

bytes in *Klass*

### 3.12.4 Class1

**class** `psd_tools.psd.descriptor.Class1` (*name: str = "", classID: bytes = b'x00x00x00x00'*)

Class structure equivalent to *Class*.

### 3.12.5 Class2

**class** `psd_tools.psd.descriptor.Class2` (*name: str = "*, *classID: bytes = b'x00x00x00x00'*)  
Class structure equivalent to *Class*.

### 3.12.6 Class3

**class** `psd_tools.psd.descriptor.Class3` (*name: str = "*, *classID: bytes = b'x00x00x00x00'*)  
Class structure equivalent to *Class*.

### 3.12.7 Descriptor

**class** `psd_tools.psd.descriptor.Descriptor` (*items=NOTHING*, *name: str = "*, *classID=b'null'*)

Dict-like descriptor structure.

Key values can be 4-character *bytes* in *Key* or arbitrary length *bytes*. Supports direct access by *Key*.

Example:

```
from psd_tools.terminology import Key

descriptor[Key.Enabled]

for key in descriptor:
    print(descriptor[key])
```

**name**  
*str*

**classID**  
*bytes* in *Klass*

### 3.12.8 Double

**class** `psd_tools.psd.descriptor.Double` (*value=0.0*)  
Double structure.

**value**  
*float* value

### 3.12.9 Enumerated

**class** `psd_tools.psd.descriptor.Enumerated` (*typeID: bytes = b'x00x00x00x00'*, *enum: bytes = b'x00x00x00x00'*)

Enum structure.

**typeID**  
*bytes* in *Type*

**enum**  
*bytes* in *Enum*

**get\_name()**  
Get enum name.

### 3.12.10 EnumeratedReference

```
class psd_tools.psd.descriptor.EnumeratedReference (name: str = "", classID: bytes =
b'x00x00x00x00', typeID: bytes =
b'x00x00x00x00', enum: bytes =
b'x00x00x00x00')
```

Enumerated reference structure.

```
name
    str value

classID
    bytes in Klass

typeID
    bytes in Type

enum
    bytes in Enum
```

### 3.12.11 GlobalObject

```
class psd_tools.psd.descriptor.GlobalObject (items=NOTHING, name: str = "", clas-
sID=b'null')
```

Global object structure equivalent to *Descriptor*.

### 3.12.12 Identifier

```
class psd_tools.psd.descriptor.Identifier (value=0)
```

Identifier equivalent to *Integer*.

### 3.12.13 Index

```
class psd_tools.psd.descriptor.Index (value=0)
```

Index equivalent to *Integer*.

### 3.12.14 Integer

```
class psd_tools.psd.descriptor.Integer (value=0)
```

Integer structure.

```
value
    int value
```

### 3.12.15 LargeInteger

```
class psd_tools.psd.descriptor.LargeInteger (value=0)
```

LargeInteger structure.

```
value
    int value
```

### 3.12.16 List

**class** `psd_tools.psd.descriptor.List` (*items=NOTHING*)  
List structure.

Example:

```
for item in list_value:  
    print(item)
```

### 3.12.17 Name

**class** `psd_tools.psd.descriptor.Name` (*name: str = "", classID: bytes = b'x00x00x00x00', value: str = ""*)  
Name structure (Undocumented).

**name**  
str

**classID**  
bytes in *Klass*

**value**  
str

### 3.12.18 ObjectArray

**class** `psd_tools.psd.descriptor.ObjectArray` (*items=NOTHING, items\_count: int = 0, name: str = "", classID=b'null'*)  
Object array structure almost equivalent to *Descriptor*.

**items\_count**  
*int* value

**name**  
*str* value

**classID**  
bytes in *Klass*

### 3.12.19 Property

**class** `psd_tools.psd.descriptor.Property` (*name: str = "", classID: bytes = b'x00x00x00x00', keyID: bytes = b'x00x00x00x00'*)  
Property structure.

**name**  
*str* value

**classID**  
bytes in *Klass*

**keyID**  
bytes in *Key*

### 3.12.20 Offset

```
class psd_tools.psd.descriptor.Offset (name: str = "", classID: bytes = b'x00x00x00x00',
                                         value=0)
```

Offset structure.

**name**  
*str* value

**classID**  
bytes in *Klass*

**value**  
*int* value

### 3.12.21 Path

```
class psd_tools.psd.descriptor.Path (value: bytes = b'x00x00x00x00')
```

Undocumented path structure equivalent to *RawData*.

### 3.12.22 RawData

```
class psd_tools.psd.descriptor.RawData (value: bytes = b'x00x00x00x00')
```

RawData structure.

**value**  
*bytes* value

### 3.12.23 Reference

```
class psd_tools.psd.descriptor.Reference (items=NOTHING)
```

Reference structure equivalent to *List*.

### 3.12.24 String

```
class psd_tools.psd.descriptor.String (value: str = "")
```

String structure.

**value**  
*str* value

### 3.12.25 UnitFloat

```
class psd_tools.psd.descriptor.UnitFloat (value: float = 0.0, unit=<Unit._None:
                                             b'#Nne'>)
```

Unit float structure.

**unit**  
unit of the value in Unit

**value**  
*float* value

### 3.12.26 UnitFloats

```
class psd_tools.psd.descriptor.UnitFloats (unit=<Unit._None: b'#Nne'>, values=NOTHING)
```

Unit floats structure.

**unit**

unit of the value in Unit

**values**

List of *float* values

## 3.13 psd\_tools.psd.engine\_data

EngineData structure.

PSD file embeds text formatting data in its own markup language referred EngineData. The format looks like the following:

```
<<
/EngineDict
<<
  /Editor
  <<
    /Text (~Make a change and save.)
  >>
>>
/Font
<<
  /Name (~HelveticaNeue-Light)
  /FillColor
  <<
    /Type 1
    /Values [ 1.0 0.0 0.0 0.0 ]
  >>
  /StyleSheetSet [
  <<
    /Name (~Normal RGB)
  >>
  ]
>>
>>
```

### 3.13.1 EngineData

```
class psd_tools.psd.engine_data.EngineData (items=NOTHING)
```

Dict-like element.

TYPE\_TOOL\_OBJECT\_SETTING tagged block contains this object in its descriptor.

### 3.13.2 EngineData2

```
class psd_tools.psd.engine_data.EngineData2 (items=NOTHING)
```

Dict-like element.



TEXT\_ENGINE\_DATA tagged block has this object.

### 3.13.3 Bool

**class** `psd_tools.psd.engine_data.Bool` (*value=False*)  
Bool element.

### 3.13.4 Dict

**class** `psd_tools.psd.engine_data.Dict` (*items=NOTHING*)  
Dict-like element.

### 3.13.5 Float

**class** `psd_tools.psd.engine_data.Float` (*value=0.0*)  
Float element.

### 3.13.6 Integer

**class** `psd_tools.psd.engine_data.Integer` (*value=0*)  
Integer element.

### 3.13.7 List

**class** `psd_tools.psd.engine_data.List` (*items=NOTHING*)  
List-like element.

### 3.13.8 Property

**class** `psd_tools.psd.engine_data.Property` (*value=None*)  
Property element.

### 3.13.9 String

**class** `psd_tools.psd.engine_data.String` (*value=None*)  
String element.

## 3.14 `psd_tools.psd.effects_layer`

Effects layer structure.

Note the structures in this module is obsolete and object-based layer effects are stored in tagged blocks.

### 3.14.1 EffectsLayer

**class** `psd_tools.psd.effects_layer.EffectsLayer` (*items=NOTHING, version: int = 0*)  
Dict-like EffectsLayer structure. See `psd_tools.constants.EffectOSType` for available keys.

**version**

### 3.14.2 CommonStateInfo

**class** `psd_tools.psd.effects_layer.CommonStateInfo` (*version: int = 0, visible: int = 1*)  
Effects layer common state info.

**version**

**visible**

### 3.14.3 ShadowInfo

**class** `psd_tools.psd.effects_layer.ShadowInfo` (*version: int = 0, blur: int = 0, intensity: int = 0, angle: int = 0, distance: int = 0, color=NOTHING, blend\_mode=<BlendMode.NORMAL: b'norm'>, enabled: int = 0, use\_global\_angle: int = 0, opacity: int = 0, native\_color=NOTHING*)

Effects layer shadow info.

**version**

**blur**

**intensity**

**angle**

**distance**

**color**

**blend\_mode**

**enabled**

**use\_global\_angle**

**opacity**

**native\_color**

### 3.14.4 OuterGlowInfo

**class** `psd_tools.psd.effects_layer.OuterGlowInfo` (*version: int = 0, blur: int = 0, intensity: int = 0, color=NOTHING, blend\_mode=<BlendMode.NORMAL: b'norm'>, enabled: int = 0, opacity: int = 0, native\_color=None*)

Effects layer outer glow info.

**version**

```

blur
intensity
color
blend_mode
enabled
opacity
native_color

```

### 3.14.5 InnerGlowInfo

```

class psd_tools.psd.effects_layer.InnerGlowInfo (version: int = 0, blur: int = 0, in-
intensity: int = 0, color=NOTHING,
blend_mode=<BlendMode.NORMAL:
b'norm'>, enabled: int = 0, opac-
ity: int = 0, invert=None, na-
tive_color=None)

```

Effects layer inner glow info.

```

version
blur
intensity
color
blend_mode
enabled
opacity
invert
native_color

```

### 3.14.6 BevelInfo

```

class psd_tools.psd.effects_layer.BevelInfo (version: int = 0, angle: int = 0,
depth: int = 0, blur: int = 0, high-
light_blend_mode=<BlendMode.NORMAL:
b'norm'>, shadow_blend_mode=<BlendMode.NORMAL:
b'norm'>, highlight_color=NOTHING,
shadow_color=NOTHING, bevel_style:
int = 0, highlight_opacity: int = 0,
shadow_opacity: int = 0, enabled: int
= 0, use_global_angle: int = 0, direc-
tion: int = 0, real_highlight_color=None,
real_shadow_color=None)

```

Effects layer bevel info.

```

version
angle
depth

```

`blur`  
`highlight_blend_mode`  
`shadow_blend_mode`  
`highlight_color`  
`shadow_color`  
`highlight_opacity`  
`shadow_opacity`  
`enabled`  
`use_global_angle`  
`direction`  
`real_highlight_color`  
`real_shadow_color`

### 3.14.7 SolidFillInfo

`class psd_tools.psd.effects_layer.SolidFillInfo` (*version: int = 2, blend\_mode=<BlendMode.NORMAL: b'norm'>, color=NOTHING, opacity: int = 0, enabled: int = 0, native\_color=NOTHING*)

Effects layer inner glow info.

`version`  
`blend_mode`  
`color`  
`opacity`  
`enabled`  
`native_color`

## 3.15 psd\_tools.psd.filter\_effects

Filter effects structure.

### 3.15.1 FilterEffects

`class psd_tools.psd.filter_effects.FilterEffects` (*items=NOTHING, version: int = 1*)  
List-like FilterEffects structure. See *FilterEffect*.  
`version`

### 3.15.2 FilterEffect

```
class psd_tools.psd.filter_effects.FilterEffect (uuid=None, version=None,  
rectangle=None, depth=None,  
max_channels=None, channels=None,  
extra=None)
```

FilterEffect structure.

```
uuid  
version  
rectangle  
depth  
max_channels  
channels  
    List of FilterEffectChannel.  
extra  
    See FilterEffectExtra.
```

### 3.15.3 FilterEffectChannel

```
class psd_tools.psd.filter_effects.FilterEffectChannel (is_written=0, compression=None, data=b'')
```

FilterEffectChannel structure.

```
is_written  
compression  
data
```

### 3.15.4 FilterEffectExtra

```
class psd_tools.psd.filter_effects.FilterEffectExtra (is_written=0, rectangle=NOTHING, compression:int = 0, data: bytes = b'')
```

FilterEffectExtra structure.

```
is_written  
rectangle  
compression  
data
```

## 3.16 psd\_tools.psd.header

File header structure.

### 3.16.1 FileHeader

**class** `psd_tools.psd.header.FileHeader` (*signature: bytes = b'8BPS', version: int = 1, channels: int = 4, height: int = 64, width: int = 64, depth: int = 8, color\_mode=<ColorMode.RGB: 3>*)

Header section of the PSD file.

Example:

```
from psd_tools.psd.header import FileHeader
from psd_tools.constants import ColorMode

header = FileHeader(channels=2, height=359, width=400, depth=8,
                    color_mode=ColorMode.GRAYSCALE)
```

**signature**

Signature: always equal to `b'8BPS'`.

**version**

Version number. PSD is 1, and PSB is 2.

**channels**

The number of channels in the image, including any user-defined alpha channel.

**height**

The height of the image in pixels.

**width**

The width of the image in pixels.

**depth**

The number of bits per channel.

**color\_mode**

The color mode of the file. See [ColorMode](#)

## 3.17 psd\_tools.psd.image\_data

Image data section structure.

*ImageData* corresponds to the last section of the PSD/PSB file where a composited image is stored. When the file does not contain layers, this is the only place pixels are saved.

### 3.17.1 ImageData

**class** `psd_tools.psd.image_data.ImageData` (*compression=<Compression.RAW: 0>, data: bytes = b''*)

Merged channel image data.

**compression**

See [Compression](#).

**data**

*bytes* as compressed in the *compression* flag.

**get\_data** (*header, split=True*)

Get decompressed data.

**Parameters header** – See *FileHeader*.

**Returns** *list* of bytes corresponding each channel.

**classmethod new** (*header*, *color=0*, *compression=<Compression.RAW: 0>*)  
Create a new image data object.

**Parameters**

- **header** – FileHeader.
- **compression** – compression type.
- **color** – default color. int or iterable for channel length.

**set\_data** (*data*, *header*)  
Set raw data and compress.

**Parameters**

- **data** – list of raw data bytes corresponding channels.
- **compression** – compression type, see *Compression*.
- **header** – See *FileHeader*.

**Returns** length of compressed data.

## 3.18 psd\_tools.psd.image\_resources

Image resources section structure. Image resources are used to store non-pixel data associated with images, such as pen tool paths or slices.

See *Resource* to check available resource names.

Example:

```
from psd_tools.constants import Resource

version_info = psd.image_resources.get_data(Resource.VERSION_INFO)
```

The following resources are plain bytes:

```
Resource.OBSOLETE1: 1000
Resource.MAC_PRINT_MANAGER_INFO: 1001
Resource.MAC_PAGE_FORMAT_INFO: 1002
Resource.OBSOLETE2: 1003
Resource.DISPLAY_INFO_OBSOLETE: 1007
Resource.BORDER_INFO: 1009
Resource.DUOTONE_IMAGE_INFO: 1018
Resource.EFFECTIVE_BW: 1019
Resource.OBSOLETE3: 1020
Resource.EPS_OPTIONS: 1021
Resource.QUICK_MASK_INFO: 1022
Resource.OBSOLETE4: 1023
Resource.WORKING_PATH: 1025
Resource.OBSOLETE5: 1027
Resource.IPTC_NAA: 1028
Resource.IMAGE_MODE_RAW: 1029
Resource.JPEG_QUALITY: 1030
Resource.URL: 1035
```

(continues on next page)

```

Resource.COLOR_SAMPLERS_RESOURCE_OBSOLETE: 1038
Resource.ICC_PROFILE: 1039
Resource.SPOT_HALFTONE: 1043
Resource.JUMP_TO_XPEP: 1052
Resource.EXIF_DATA_1: 1058
Resource.EXIF_DATA_3: 1059
Resource.XMP_METADATA: 1060
Resource.CAPTION_DIGEST: 1061
Resource.ALTERNATE_DUOTONE_COLORS: 1066
Resource.ALTERNATE_SPOT_COLORS: 1067
Resource.HDR_TONING_INFO: 1070
Resource.PRINT_INFO_CS2: 1071
Resource.COLOR_SAMPLERS_RESOURCE: 1073
Resource.DISPLAY_INFO: 1077
Resource.MAC_NSPRINTINFO: 1084
Resource.WINDOWS_DEVMODE: 1085
Resource.PATH_INFO_N: 2000-2999
Resource.PLUGIN_RESOURCES_N: 4000-4999
Resource.IMAGE_READY_VARIABLES: 7000
Resource.IMAGE_READY_DATA_SETS: 7001
Resource.IMAGE_READY_DEFAULT_SELECTED_STATE: 7002
Resource.IMAGE_READY_7_ROLLOVER_EXPANDED_STATE: 7003
Resource.IMAGE_READY_ROLLOVER_EXPANDED_STATE: 7004
Resource.IMAGE_READY_SAVE_LAYER_SETTINGS: 7005
Resource.IMAGE_READY_VERSION: 7006
Resource.LIGHTROOM_WORKFLOW: 8000

```

### 3.18.1 ImageResources

**class** `psd_tools.psd.image_resources.ImageResources` (*items=NOTHING*)  
 Image resources section of the PSD file. Dict of *ImageResource*.

**get\_data** (*key, default=None*)  
 Get data from the image resources.

Shortcut for the following:

```

if key in image_resources:
    value = tagged_blocks[key].data

```

**classmethod new** (*\*\*kwargs*)  
 Create a new default image resources.

**Returns** ImageResources

### 3.18.2 ImageResource

**class** `psd_tools.psd.image_resources.ImageResource` (*signature: bytes = b'8BIM', key: int = 1000, name: str = "", data: bytes = b"*)

Image resource block.

**signature**  
 Binary signature, always b'8BIM'.



**key**  
Unique identifier for the resource. See *Resource*.

**name**

**data**  
The resource data.

### 3.18.3 AlphaIdentifiers

**class** `psd_tools.psd.image_resources.AlphaIdentifiers` (*items=NOTHING*)  
List of alpha identifiers.

### 3.18.4 AlphaNamesPascal

**class** `psd_tools.psd.image_resources.AlphaNamesPascal` (*items=NOTHING*)  
List of alpha names.

### 3.18.5 AlphaNamesUnicode

**class** `psd_tools.psd.image_resources.AlphaNamesUnicode` (*items=NOTHING*)  
List of alpha names.

### 3.18.6 Byte

**class** `psd_tools.psd.image_resources.Byte` (*value=0*)  
Byte element.

### 3.18.7 GridGuidesInfo

**class** `psd_tools.psd.image_resources.GridGuidesInfo` (*version: int = 1, horizontal: int = 0, vertical: int = 0, data=NOTHING*)  
Grid and guides info structure.

### 3.18.8 HalftoneScreens

**class** `psd_tools.psd.image_resources.HalftoneScreens` (*items=NOTHING*)  
Halftone screens.

### 3.18.9 HalftoneScreen

**class** `psd_tools.psd.image_resources.HalftoneScreen` (*freq: int = 0, unit: int = 0, angle: int = 0, shape: int = 0, use\_accurate: bool = False, use\_printer: bool = False*)  
Halftone screen.

**freq**

`unit`  
`angle`  
`shape`  
`use_accurate`  
`use_printer`

### 3.18.10 Integer

`class psd_tools.psd.image_resources.Integer (value=0)`  
Integer element.

### 3.18.11 LayerGroupEnabledIDs

`class psd_tools.psd.image_resources.LayerGroupEnabledIDs (items=NOTHING)`  
Layer group enabled ids.

### 3.18.12 LayerGroupInfo

`class psd_tools.psd.image_resources.LayerGroupInfo (items=NOTHING)`  
Layer group info list.

### 3.18.13 LayerSelectionIDs

`class psd_tools.psd.image_resources.LayerSelectionIDs (items=NOTHING)`  
Layer selection ids.

### 3.18.14 ShortInteger

`class psd_tools.psd.image_resources.ShortInteger (value=0)`  
Short integer element.

### 3.18.15 PascalString

`class psd_tools.psd.image_resources.PascalString (value=None)`  
Pascal string element.

### 3.18.16 PixelAspectRatio

`class psd_tools.psd.image_resources.PixelAspectRatio (value=0.0, version: int = 1)`  
Pixel aspect ratio.

### 3.18.17 PrintFlags

```
class psd_tools.psd.image_resources.PrintFlags (labels: bool = False, crop_marks: bool = False, colorbars: bool = False, registration_marks: bool = False, negative: bool = False, flip: bool = False, interpolate: bool = False, caption: bool = False, print_flags=None)
```

Print flags.

### 3.18.18 PrintFlagsInfo

```
class psd_tools.psd.image_resources.PrintFlagsInfo (version: int = 0, center_crop: int = 0, bleed_width_value: int = 0, bleed_width_scale: int = 0)
```

Print flags info structure.

```
version
center_crop
bleed_width_value
bleed_width_scale
```

### 3.18.19 PrintScale

```
class psd_tools.psd.image_resources.PrintScale (style=<PrintScaleStyle.CENTERED: 0>, x: float = 0.0, y: float = 0.0, scale: float = 0.0)
```

Print scale structure.

```
style
x
y
scale
```

### 3.18.20 ResolutionInfo

```
class psd_tools.psd.image_resources.ResolutionInfo (horizontal: int = 0, horizontal_unit: int = 0, width_unit: int = 0, vertical: int = 0, vertical_unit: int = 0, height_unit: int = 0)
```

Resolution info structure.

```
horizontal
horizontal_unit
width_unit
vertical
vertical_unit
```

`height_unit`

### 3.18.21 Slices

**class** `psd_tools.psd.image_resources.Slices` (*version: int = 0, data=None*)

Slices resource.

**version**

**data**

### 3.18.22 SlicesV6

**class** `psd_tools.psd.image_resources.SlicesV6` (*bbox=NOTHING, name: str = "", items=NOTHING*)

Slices resource version 6.

**bbox**

**name**

**items**

### 3.18.23 SliceV6

**class** `psd_tools.psd.image_resources.SliceV6` (*slice\_id: int = 0, group\_id: int = 0, origin: int = 0, associated\_id=None, name: str = "", slice\_type: int = 0, bbox=NOTHING, url: str = "", target: str = "", message: str = "", alt\_tag: str = "", cell\_is\_html: bool = False, cell\_text: str = "", horizontal\_align: int = 0, vertical\_align: int = 0, alpha: int = 0, red: int = 0, green: int = 0, blue: int = 0, data=None*)

Slice element for version 6.

**slice\_id**

**group\_id**

**origin**

**associated\_id**

**name**

**slice\_type**

**bbox**

**url**

**target**

**message**

**alt\_tag**

**cell\_is\_html**

**cell\_text**

**horizontal**  
**vertical**  
**alpha**  
**red**  
**green**  
**blue**  
**data**

### 3.18.24 ThumbnailResource

**class** `psd_tools.psd.image_resources.ThumbnailResource` (*fmt: int = 0, width: int = 0, height: int = 0, row: int = 0, total\_size: int = 0, bits: int = 0, planes: int = 0, data: bytes = b"*)

Thumbnail resource structure.

**fmt**  
**width**  
**height**  
**row**  
**total\_size**  
**size**  
**bits**  
**planes**  
**data**  
**topil()**

Get PIL Image.

**Returns** PIL Image object.

### 3.18.25 ThumbnailResourceV4

**class** `psd_tools.psd.image_resources.ThumbnailResourceV4` (*fmt: int = 0, width: int = 0, height: int = 0, row: int = 0, total\_size: int = 0, bits: int = 0, planes: int = 0, data: bytes = b"*)

### 3.18.26 TransferFunctions

**class** `psd_tools.psd.image_resources.TransferFunctions` (*items=NOTHING*)  
 Transfer functions.

### 3.18.27 TransferFunction

```
class psd_tools.psd.image_resources.TransferFunction (curve=NOTHING, override:  
                                                    bool = False)  
    Transfer function
```

### 3.18.28 URLList

```
class psd_tools.psd.image_resources.URLList (items=NOTHING)  
    URL list structure.
```

### 3.18.29 URLItem

```
class psd_tools.psd.image_resources.URLItem (number: int = 0, id: int = 0, name: str = "  
    URL item.  
  
    number  
  
    id  
  
    name
```

### 3.18.30 VersionInfo

```
class psd_tools.psd.image_resources.VersionInfo (version: int = 1, has_composite: bool  
                                                    = False, writer: str = "  
                                                    ", reader: str =  
                                                    ", file_version: int = 1)  
  
    Version info structure.  
  
    version  
  
    has_composite  
  
    writer  
  
    reader  
  
    file_version
```

## 3.19 psd\_tools.psd.layer\_and\_mask

Layer and mask data structure.

### 3.19.1 LayerAndMaskInformation

```
class psd_tools.psd.layer_and_mask.LayerAndMaskInformation (layer_info=None,  
                                                            global_layer_mask_info=None,  
                                                            tagged_blocks=None)  
  
    Layer and mask information section.  
  
    layer_info  
        See LayerInfo.
```

**global\_layer\_mask\_info**

See *GlobalLayerMaskInfo*.

**tagged\_blocks**

See *TaggedBlocks*.

### 3.19.2 LayerInfo

**class** `psd_tools.psd.layer_and_mask.LayerInfo` (*layer\_count: int = 0, layer\_records=None, channel\_image\_data=None*)

High-level organization of the layer information.

**layer\_count**

Layer count. If it is a negative number, its absolute value is the number of layers and the first alpha channel contains the transparency data for the merged result.

**layer\_records**

Information about each layer. See *LayerRecords*.

**channel\_image\_data**

Channel image data. See *ChannelImageData*.

### 3.19.3 GlobalLayerMaskInfo

**class** `psd_tools.psd.layer_and_mask.GlobalLayerMaskInfo` (*overlay\_color=None, opacity: int = 0, kind=<GlobalLayerMaskKind.PER\_LAYER: 128>*)

Global mask information.

**overlay\_color**

Overlay color space (undocumented) and color components.

**opacity**

Opacity. 0 = transparent, 100 = opaque.

**kind**

Kind. 0 = Color selected–i.e. inverted; 1 = Color protected; 128 = use value stored per layer. This value is preferred. The others are for backward compatibility with beta versions.

### 3.19.4 LayerRecords

**class** `psd_tools.psd.layer_and_mask.LayerRecords` (*items=NOTHING*)

List of layer records. See *LayerRecord*.

### 3.19.5 LayerRecord

```
class psd_tools.psd.layer_and_mask.LayerRecord (top: int = 0, left: int = 0, bottom: int = 0,  
right: int = 0, channel_info=NOTHING,  
signature: bytes = b'8BIM',  
blend_mode=<BlendMode.NORMAL:  
b'norm'>, opacity: int = 255,  
clipping=<Clipping.BASE: 0>,  
flags=NOTHING, mask_data=None,  
blending_ranges=NOTHING, name: str  
= "", tagged_blocks=NOTHING)
```

Layer record.

**top**

Top position.

**left**

Left position.

**bottom**

Bottom position.

**right**

Right position.

**channel\_info**

List of *ChannelInfo*.

**signature**

Blend mode signature b'8BIM'.

**blend\_mode**

Blend mode key. See *BlendMode*.

**opacity**

Opacity, 0 = transparent, 255 = opaque.

**clipping**

Clipping, 0 = base, 1 = non-base. See *Clipping*.

**flags**

See *LayerFlags*.

**mask\_data**

*MaskData* or None.

**blending\_ranges**

See *LayerBlendingRanges*.

**name**

Layer name.

**tagged\_blocks**

See *TaggedBlocks*.

**channel\_sizes**

List of channel sizes: [(width, height)].

**height**

Height of the layer.

**width**

Width of the layer.



### 3.19.6 LayerFlags

```
class psd_tools.psd.layer_and_mask.LayerFlags (transparency_protected: bool = False,  
visible: bool = True, obsolete: bool = False, photoshop_v5_later: bool =  
True, pixel_data_irrelevant: bool = False, undocumented_1: bool = False, un-  
documented_2: bool = False, undocu-  
mented_3: bool = False)
```

Layer flags.

Note there are undocumented flags. Maybe photoshop version.

**transparency\_protected**

**visible**

**pixel\_data\_irrelevant**

### 3.19.7 LayerBlendingRanges

```
class psd_tools.psd.layer_and_mask.LayerBlendingRanges (composite_ranges=NOTHING,  
chan-  
nel_ranges=NOTHING)
```

Layer blending ranges.

All ranges contain 2 black values followed by 2 white values.

**composite\_ranges**

List of composite gray blend source and destination ranges.

**channel\_ranges**

List of channel source and destination ranges.

### 3.19.8 MaskData

```
class psd_tools.psd.layer_and_mask.MaskData (top: int = 0, left: int = 0, bot-  
tom: int = 0, right: int = 0, back-  
ground_color: int = 0, flags=NOTHING,  
parameters=None, real_flags=None,  
real_background_color=None,  
real_top=None, real_left=None,  
real_bottom=None, real_right=None)
```

Mask data.

Real user mask is a final composite mask of vector and pixel masks.

**top**

Top position.

**left**

Left position.

**bottom**

Bottom position.

**right**

Right position.

**background\_color**

Default color. 0 or 255.

**flags**

See *MaskFlags*.

**parameters**

*MaskParameters* or None.

**real\_flags**

Real user mask flags. See *MaskFlags*.

**real\_background\_color**

Real user mask background. 0 or 255.

**real\_top**

Top position of real user mask.

**real\_left**

Left position of real user mask.

**real\_bottom**

Bottom position of real user mask.

**real\_right**

Right position of real user mask.

**height**

Height of the mask.

**real\_height**

Height of real user mask.

**real\_width**

Width of real user mask.

**width**

Width of the mask.

### 3.19.9 MaskFlags

```
class psd_tools.psd.layer_and_mask.MaskFlags (pos_relative_to_layer: bool = False,  
mask_disabled: bool = False, invert_mask:  
bool = False, user_mask_from_render:  
bool = False, parameters_applied: bool  
= False, undocumented_1: bool = False,  
undocumented_2: bool = False, undocu-  
mented_3: bool = False)
```

Mask flags.

**pos\_relative\_to\_layer**

Position relative to layer.

**mask\_disabled**

Layer mask disabled.

**invert\_mask**

Invert layer mask when blending (Obsolete).

**user\_mask\_from\_render**

The user mask actually came from rendering other data.

**parameters\_applied**

The user and/or vector masks have parameters applied to them.

**3.19.10 MaskParameters**

```
class psd_tools.psd.layer_and_mask.MaskParameters (user_mask_density=None,
                                                    user_mask_feather=None,   vec-
                                                    tor_mask_density=None,   vec-
                                                    tor_mask_feather=None)
```

Mask parameters.

**user\_mask\_density**

**user\_mask\_feather**

**vector\_mask\_density**

**vector\_mask\_feather**

**3.19.11 ChannelInfo**

```
class psd_tools.psd.layer_and_mask.ChannelInfo (id=<ChannelID.CHANNEL_0: 0>,
                                                    length: int = 0)
```

Channel information.

**id**

Channel ID: 0 = red, 1 = green, etc.; -1 = transparency mask; -2 = user supplied layer mask, -3 real user supplied layer mask (when both a user mask and a vector mask are present). See [ChannelID](#).

**length**

Length of the corresponding channel data.

**3.19.12 ChannelImageData**

```
class psd_tools.psd.layer_and_mask.ChannelImageData (items=NOTHING)
```

List of channel data list.

This size of this list corresponds to the size of [LayerRecords](#). Each item corresponds to the channels of each layer.

See [ChannelDataList](#).

**3.19.13 ChannelDataList**

```
class psd_tools.psd.layer_and_mask.ChannelDataList (items=NOTHING)
```

List of channel image data, corresponding to each color or alpha.

See [ChannelData](#).

**3.19.14 ChannelData**

```
class psd_tools.psd.layer_and_mask.ChannelData (compression=<Compression.RAW: 0>,
                                                    data: bytes = b")
```

Channel data.

**compression**

Compression type. See *Compression*.

**data**

Data.

**get\_data** (*width, height, depth, version=1*)

Get decompressed channel data.

**Parameters**

- **width** – width.
- **height** – height.
- **depth** – bit depth of the pixel.
- **version** – psd file version.

**Return type** bytes

**set\_data** (*data, width, height, depth, version=1*)

Set raw channel data and compress to store.

**Parameters**

- **data** – raw data bytes to write.
- **compression** – compression type, see *Compression*.
- **width** – width.
- **height** – height.
- **depth** – bit depth of the pixel.
- **version** – psd file version.

## 3.20 psd\_tools.psd.linked\_layer

Linked layer structure.

### 3.20.1 LinkedLayers

**class** `psd_tools.psd.linked_layer.LinkedLayers` (*items=NOTHING*)

List of `LinkedLayer` structure. See *LinkedLayer*.

### 3.20.2 LinkedLayer

**class** `psd_tools.psd.linked_layer.LinkedLayer` (*kind=<LinkedLayerType.ALIAS: b'liFA'>, version=1, uuid: str = "", filename: str = "", filetype: bytes = b'x00x00x00x00', creator: bytes = b'x00x00x00x00', filesize=None, open\_file=None, linked\_file=None, timestamp=None, data=None, child\_id=None, mod\_time=None, lock\_state=None*)

`LinkedLayer` structure.

**kind**

```

version
uuid
filename
filetype
creator
filesize
open_file
linked_file
timestamp
data
child_id
mod_time
lock_state

```

## 3.21 psd\_tools.psd.patterns

Patterns structure.

### 3.21.1 Patterns

**class** `psd_tools.psd.patterns.Patterns` (*items=NOTHING*)  
 List of Pattern structure. See *Pattern*.

### 3.21.2 Pattern

**class** `psd_tools.psd.patterns.Pattern` (*version: int = 1, image\_mode=<enum 'ColorMode'>, point=None, name: str = "", pattern\_id: str = "", color\_table=None, data=None*)

Pattern structure.

```

version
image_mode
    See ColorMode
point
    Size in tuple.
name
    str name of the pattern.
pattern_id
    ID of this pattern.
color_table
    Color table if the mode is INDEXED.

```

**data**

See *VirtualMemoryArrayList*

### 3.21.3 VirtualMemoryArrayList

**class** `psd_tools.psd.patterns.VirtualMemoryArrayList` (*version: int = 3, rectangle=None, channels=None*)

VirtualMemoryArrayList structure. Container of channels.

**version**

**rectangle**

Tuple of *int*

**channels**

List of *VirtualMemoryArray*

### 3.21.4 VirtualMemoryArray

**class** `psd_tools.psd.patterns.VirtualMemoryArray` (*is\_written=0, depth=None, rectangle=None, pixel\_depth=None, compression=<Compression.RAW: 0>, data=b''*)

VirtualMemoryArrayList structure, corresponding to each channel.

**is\_written**

**depth**

**rectangle**

**pixel\_depth**

**compression**

**data**

**get\_data()**

Get decompressed bytes.

**set\_data** (*size, data, depth, compression=0*)

Set bytes.

## 3.22 psd\_tools.psd.tagged\_blocks

Tagged block data structure.

---

**Todo:** Support the following tagged blocks: `Tag.PATTERN_DATA`, `Tag.TYPE_TOOL_INFO`, `Tag.LAYER`, `Tag.ALPHA`

---

### 3.22.1 TaggedBlocks

**class** `psd_tools.psd.tagged_blocks.TaggedBlocks` (*items=NOTHING*)

Dict of tagged block items.

See *Tag* for available keys.

Example:

```
from psd_tools.constants import Tag

# Iterate over fields
for key in tagged_blocks:
    print(key)

# Get a field
value = tagged_blocks.get_data(Tag, TYPE_TOOL_OBJECT_SETTING)
```

### 3.22.2 TaggedBlock

**class** `psd_tools.psd.tagged_blocks.TaggedBlock` (*signature=b'8BIM', key=b'', data=b''*)

Layer tagged block with extra info.

**key**

4-character code. See *Tag*

**data**

Data.

### 3.22.3 Annotations

**class** `psd_tools.psd.tagged_blocks.Annotations` (*items=NOTHING, major\_version: int = 2, minor\_version: int = 1*)

List of Annotation, see `:py:class: .Annotation`.

**major\_version**

**minor\_version**

### 3.22.4 Annotation

**class** `psd_tools.psd.tagged_blocks.Annotation` (*kind: bytes = b'txtA', is\_open: int = 0, flags: int = 0, optional\_blocks: int = 1, icon\_location=NOTHING, popup\_location=NOTHING, color=NOTHING, author: str = "", name: str = "", mod\_date: str = "", marker: bytes = b'txtC', data: bytes = b''*)

Annotation structure.

**kind**

**is\_open**

### 3.22.5 Bytes

**class** `psd_tools.psd.tagged_blocks.Bytes` (*value: bytes = b'x00x00x00x00'*)

Bytes structure.

**value**

### 3.22.6 ChannelBlendingRestrictionsSetting

**class** `psd_tools.psd.tagged_blocks.ChannelBlendingRestrictionsSetting` (*items=NOTHING*)  
ChannelBlendingRestrictionsSetting structure.

List of restricted channel numbers (int).

### 3.22.7 FilterMask

**class** `psd_tools.psd.tagged_blocks.FilterMask` (*color=None, opacity: int = 0*)  
FilterMask structure.

**color**

**opacity**

### 3.22.8 MetadataSettings

**class** `psd_tools.psd.tagged_blocks.MetadataSettings` (*items=NOTHING*)  
MetadataSettings structure.

### 3.22.9 MetadataSetting

**class** `psd_tools.psd.tagged_blocks.MetadataSetting` (*signature: bytes = b'8BIM', key: bytes = b'', copy\_on\_sheet: bool = False, data: bytes = b''*)

MetadataSetting structure.

### 3.22.10 PixelSourceData2

**class** `psd_tools.psd.tagged_blocks.PixelSourceData2` (*items=NOTHING*)  
PixelSourceData2 structure.

### 3.22.11 PlacedLayerData

**class** `psd_tools.psd.tagged_blocks.PlacedLayerData` (*kind: bytes = b'plcL', version: int = 3, uuid: bytes = "", page: int = 0, total\_pages: int = 0, anti\_alias: int = 0, layer\_type=<PlacedLayerType.UNKNOWN: 0>, transform: tuple = (0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0), warp=None*)

PlacedLayerData structure.

### 3.22.12 ProtectedSetting

**class** `psd_tools.psd.tagged_blocks.ProtectedSetting` (*value=0*)  
ProtectedSetting structure.



### 3.22.13 ReferencePoint

**class** `psd_tools.psd.tagged_blocks.ReferencePoint` (*items=NOTHING*)  
ReferencePoint structure.

### 3.22.14 SectionDividerSetting

**class** `psd_tools.psd.tagged_blocks.SectionDividerSetting` (*kind=<SectionDivider.OTHER: 0>*, *signature=None*, *blend\_mode=None*, *sub\_type=None*)

SectionDividerSetting structure.

**kind**  
**blend\_mode**  
**sub\_type**

### 3.22.15 SheetColorSetting

**class** `psd_tools.psd.tagged_blocks.SheetColorSetting` (*value=0*)  
SheetColorSetting value.

This setting represents color label in the layers panel in Photoshop UI.

**value**

### 3.22.16 SmartObjectLayerData

**class** `psd_tools.psd.tagged_blocks.SmartObjectLayerData` (*kind: bytes = b'soLD'*, *version: int = 5*, *data: psd\_tools.psd.descriptor.DescriptorBlock = None*)

VersionedDescriptorBlock structure.

**kind**  
**version**  
**data**

### 3.22.17 TypeToolObjectSetting

```
class psd_tools.psd.tagged_blocks.TypeToolObjectSetting (version: int = 1, transform: tuple = (0.0, 0.0, 0.0, 0.0, 0.0, 0.0), text_version: int = 1, text_data: psd_tools.psd.descriptor.DescriptorBlock = None, warp_version: int = 1, warp: psd_tools.psd.descriptor.DescriptorBlock = None, left: int = 0, top: int = 0, right: int = 0, bottom: int = 0)
```

TypeToolObjectSetting structure.

**version**

**transform**

Tuple of affine transform parameters (xx, xy, yx, yy, tx, ty).

**text\_version**

**text\_data**

**warp\_version**

**warp**

**left**

**top**

**right**

**bottom**

### 3.22.18 UserMask

```
class psd_tools.psd.tagged_blocks.UserMask (color=None, opacity: int = 0, flag: int = 128)  
UserMask structure.
```

**color**

**opacity**

**flag**

## 3.23 psd\_tools.psd.vector

Vector mask, path, and stroke structure.

### 3.23.1 Path

```
class psd_tools.psd.vector.Path (items=NOTHING)
```

List-like Path structure. Elements are either PathFillRule, InitialFillRule, ClipboardRecord, ClosedPath, or OpenPath.

### 3.23.2 Subpath

**class** `psd_tools.psd.vector.Subpath` (*items=NOTHING, operation: int = 1, unknown1: int = 1, unknown2: int = 0, index: int = 0, unknown3: bytes = b'x00x00x00x00x00x00x00x00x00x00'*)

Subpath element. This is a list of Knot objects.

---

**Note:** There are undocumented data associated with this structure.

---

**operation**

*int* value indicating how multiple subpath should be combined:

1: Or (union), 2: Not-Or, 3: And (intersect), 0: Xor (exclude)

The first path element is applied to the background surface. Intersection does not have strokes.

**index**

*int* index that specifies corresponding origination object.

**is\_closed()**

Returns whether if the path is closed or not.

**Returns** *bool*.

### 3.23.3 Knot

**class** `psd_tools.psd.vector.Knot` (*preceding: tuple = (0.0, 0.0), anchor: tuple = (0.0, 0.0), leaving: tuple = (0.0, 0.0)*)

Knot element consisting of 3 control points for Bezier curves.

**preceding**

(y, x) tuple of preceding control point in relative coordinates.

**anchor**

(y, x) tuple of anchor point in relative coordinates.

**leaving**

(y, x) tuple of leaving control point in relative coordinates.

### 3.23.4 ClipboardRecord

**class** `psd_tools.psd.vector.ClipboardRecord` (*top: int = 0, left: int = 0, bottom: int = 0, right: int = 0, resolution: int = 0*)

Clipboard record.

**top**

Top position in *int*

**left**

Left position in *int*

**bottom**

Bottom position in *int*

**right**

Right position in *int*

**resolution**Resolution in *int*

### 3.23.5 PathFillRule

**class** `psd_tools.psd.vector.PathFillRule`

Path fill rule record, empty.

### 3.23.6 InitialFillRule

**class** `psd_tools.psd.vector.InitialFillRule` (*value=0*)

Initial fill rule record.

**rule**

A value of 1 means that the fill starts with all pixels. The value will be either 0 or 1.

### 3.23.7 VectorMaskSetting

**class** `psd_tools.psd.vector.VectorMaskSetting` (*version: int = 3, flags: int = 0, path=None*)  
VectorMaskSetting structure.**version****path**List of *Subpath* objects.**disable**

Flag to indicate that the vector mask is disabled.

**invert**

Flag to indicate that the vector mask is inverted.

**not\_link**

Flag to indicate that the vector mask is not linked.

### 3.23.8 VectorStrokeContentSetting

**class** `psd_tools.psd.vector.VectorStrokeContentSetting` (*items=NOTHING, name: str = "", classID=b'null', key: bytes = b'x00x00x00x00', version: int = 1*)Dict-like Descriptor-based structure. See *Descriptor*.**key****version**

## 3.24 psd\_tools.terminology

Constants for descriptor.

This file is automatically generated by `tools/extract_terminology.py`

### 3.24.1 Klass

```
class psd_tools.terminology.Klass
    Klass definitions extracted from PITerminology.h.

    See https://www.adobe.com/devnet/photoshop/sdk.html

    Action = b'Actn'
    ActionSet = b'ASet'
    Adjustment = b'Adjs'
    AdjustmentLayer = b'AdjL'
    AirbrushTool = b'AbTl'
    AlphaChannelOptions = b'ACHl'
    AntiAliasedPICTAcquire = b'AntA'
    Application = b'capp'
    Arrowhead = b'cArw'
    ArtHistoryBrushTool = b'ABTl'
    Assert = b'Asrt'
    AssumedProfile = b'AssP'
    BMPFormat = b'BMPF'
    BackLight = b'BakL'
    BackgroundEraserTool = b'SETl'
    BackgroundLayer = b'BckL'
    BevelEmboss = b'ebbl'
    BitmapMode = b'BtmM'
    BlendRange = b'Blnd'
    BlurTool = b'BlTl'
    BookColor = b'BkCl'
    BrightnessContrast = b'BrgC'
    Brush = b'Brsh'
    BurnInTool = b'BrTl'
    CMYKColor = b'CMYC'
    CMYKColorMode = b'CMYM'
    CMYKSetup = b'CMYS'
    CachePrefs = b'CchP'
    Calculation = b'Clcl'
    Channel = b'Chnl'
    ChannelMatrix = b'ChMx'
    ChannelMixer = b'ChnM'
```

```
ChromeFX = b'ChFX'  
CineonFormat = b'SDPX'  
ClippingInfo = b'Clpo'  
ClippingPath = b'ClpP'  
CloneStampTool = b'ClTl'  
Color = b'Clr '  
ColorBalance = b'ClrB'  
ColorCast = b'ColC'  
ColorCorrection = b'ClrC'  
ColorPickerPrefs = b'ClrK'  
ColorSampler = b'ClSm'  
ColorStop = b'Clrt'  
Command = b'Cmd'  
Contour = b'FxSc'  
CurvePoint = b'CrPt'  
Curves = b'Crvs'  
CurvesAdjustment = b'CrvA'  
CustomPalette = b'Cstl'  
CustomPhosphors = b'CstP'  
CustomWhitePoint = b'CstW'  
DicomFormat = b'Dicm'  
DisplayPrefs = b'DspP'  
Document = b'Dcmn'  
DodgeTool = b'DdTl'  
DropShadow = b'DrSh'  
DuotoneInk = b'DtnI'  
DuotoneMode = b'DtnM'  
EPSGenericFormat = b'EPSG'  
EPSPICTPreview = b'EPSC'  
EPSTIFFPreview = b'EPST'  
EXRf = b'EXRf'  
Element = b'Elmn'  
Ellipse = b'Elps'  
EraserTool = b'ErTl'  
Export = b'Expr'  
FileInfo = b'FlIn'
```

```
FileSavePrefs = b'FlSv'  
FillFlash = b'FilF'  
FlashPixFormat = b'FlsP'  
FontDesignAxes = b'FntD'  
Format = b'Fmt '  
FrameFX = b'FrFX'  
GIF89aExport = b'GF89'  
GIFFormat = b'GFFr'  
GeneralPrefs = b'GnrP'  
GlobalAngle = b'gblA'  
Gradient = b'Grdn'  
GradientFill = b'Grdf'  
GradientMap = b'GdMp'  
GradientTool = b'GrTl'  
GraySetup = b'GrSt'  
Grayscale = b'Grsc'  
GrayscaleMode = b'Grys'  
Guide = b'Gd '  
GuidesPrefs = b'GdPr'  
HSBColor = b'HSBC'  
HSBColorMode = b'HSBM'  
HalftoneScreen = b'Hlfs'  
HalftoneSpec = b'Hlfp'  
HistoryBrushTool = b'HBTl'  
HistoryPrefs = b'CHsP'  
HistoryState = b'HstS'  
HueSatAdjustment = b'HStA'  
HueSatAdjustmentV2 = b'Hst2'  
HueSaturation = b'HStr'  
IFFFormat = b'IFFF'  
IllustratorPathsExport = b'IlsP'  
ImagePoint = b'ImgP'  
Import = b'Impr'  
IndexedColorMode = b'IndC'  
InkTransfer = b'InkT'  
InnerGlow = b'IrGl'
```

```
InnerShadow = b'IrSh'  
InterfaceColor = b'IClr'  
Invert = b'Invr'  
JPEGFormat = b'JPEG'  
LabColor = b'LbCl'  
LabColorMode = b'LbCM'  
Layer = b'Lyr '  
LayerEffects = b'Lefx'  
LayerFXVisible = b'lfxv'  
Levels = b'Lvls'  
LevelsAdjustment = b'LvlA'  
LightSource = b'LghS'  
Line = b'Ln '  
MacPaintFormat = b'McPn'  
MagicEraserTool = b'MgEr'  
MagicPoint = b'Mgcp'  
Mask = b'Msk '  
MenuItem = b'Mn '  
Mode = b'Md '  
MultichannelMode = b'MltC'  
Null = b'null'  
ObsoleteTextLayer = b'TxLy'  
Offset = b'Ofst'  
Opacity = b'Opac'  
OuterGlow = b'OrGl'  
PDFGenericFormat = b'PDFG'  
PICTFileFormat = b'PICF'  
PICTResourceFormat = b'PICR'  
PNGFormat = b'PNGF'  
PageSetup = b'PgSt'  
PaintbrushTool = b'PbTl'  
Path = b'Path'  
PathComponent = b'PaCm'  
PathPoint = b'Pthp'  
Pattern = b'PttR'  
PatternStampTool = b'PaTl'
```



```
PencilTool = b'PcTl'  
Photoshop20Format = b'Pht2'  
Photoshop35Format = b'Pht3'  
PhotoshopDCS2Format = b'PhD2'  
PhotoshopDCSFormat = b'PhD1'  
PhotoshopEPSFormat = b'PhtE'  
PhotoshopPDFFormat = b'PhtP'  
Pixel = b'Pxel'  
PixelPaintFormat = b'Px1P'  
PluginPrefs = b'PlgP'  
Point = b'Pnt '  
Point16 = b'Pnt1'  
Polygon = b'Plgn'  
Posterize = b'Pstr'  
Preferences = b'GnrP'  
ProfileSetup = b'Prfs'  
Property = b'Prpr'  
RGBColor = b'RGBC'  
RGBColorMode = b'RGBM'  
RGBSetup = b'RGBt'  
Range = b'Rang'  
RawFormat = b'Rw '  
Rect16 = b'Rct1'  
Rectangle = b'Rctn'  
SaturationTool = b'SrTl'  
ScitexCTFormat = b'Sctx'  
Selection = b'csel'  
SelectiveColor = b'SlcC'  
ShapingCurve = b'ShpC'  
SharpenTool = b'ShTl'  
SingleColumn = b'Sngc'  
SingleRow = b'Sngr'  
SmudgeTool = b'SmTl'  
Snapshot = b'SnpS'  
SolidFill = b'SoFi'  
SpotColorChannel = b'SCch'
```



```
ADSVERTICAL = b'Advr'  
ASCII = b'ASCI'  
AboutApp = b'AbAp'  
AbsColorimetric = b'AClr'  
Absolute = b'Absl'  
ActualPixels = b'ActP'  
Adaptive = b'Adpt'  
Add = b'Add '  
AdjustmentOptions = b'AdjO'  
AdobeRGB1998 = b'SMPT'  
AirbrushEraser = b'Arbs'  
All = b'Al '  
Amiga = b'Amga'  
AmountHigh = b'amHi'  
AmountLow = b'amLo'  
AmountMedium = b'amMd'  
Angle = b'Angl'  
AntiAliasCrisp = b'AnCr'  
AntiAliasHigh = b'AnHi'  
AntiAliasLow = b'AnLo'  
AntiAliasMedium = b'AnMd'  
AntiAliasNone = b'Anno'  
AntiAliasSmooth = b'AnSm'  
AntiAliasStrong = b'AnSt'  
Any = b'Any '  
AppleRGB = b'AppR'  
ApplyImage = b'AplI'  
AroundCenter = b'ArnC'  
Arrange = b'Arng'  
Ask = b'Ask '  
AskWhenOpening = b'AskW'  
B = b'B '  
Back = b'Back'  
Background = b'Bckg'  
BackgroundColor = b'BckC'  
Backward = b'Bckw'
```

Behind = b'Bhnd'  
Best = b'Bst '  
Better = b'Dthb'  
Bicubic = b'Bcbc'  
Bilinear = b'Blnr'  
Binary = b'Brny'  
BitDepth1 = b'BD1 '  
BitDepth16 = b'BD16'  
BitDepth24 = b'BD24'  
BitDepth32 = b'BD32'  
BitDepth4 = b'BD4 '  
BitDepth8 = b'BD8 '  
BitDepthA1R5G5B5 = b'1565'  
BitDepthA4R4G4B4 = b'4444'  
BitDepthR5G6B5 = b'x565'  
BitDepthX4R4G4B4 = b'x444'  
BitDepthX8R8G8B8 = b'x888'  
Bitmap = b'Btmp'  
Black = b'Blck'  
BlackAndWhite = b'BanW'  
BlackBody = b'BlcB'  
Blacks = b'Blks'  
Blast = b'Blst'  
BlockEraser = b'Blk '  
Blocks = b'Blks'  
Blue = b'Bl '  
Blues = b'BlS '  
Bottom = b'Bttm'  
BrushDarkRough = b'BrDR'  
BrushLightRough = b'BrSL'  
BrushSimple = b'BrSm'  
BrushSize = b'BrSs'  
BrushSparkle = b'BrSp'  
BrushWideBlurry = b'BrbW'  
BrushWideSharp = b'BrSW'  
BrushesAppend = b'BrSA'

```
BrushesDefine = b'BrsD'  
BrushesDelete = b'Brsf'  
BrushesLoad = b'Brsd'  
BrushesNew = b'BrsN'  
BrushesOptions = b'BrsO'  
BrushesReset = b'BrsR'  
BrushesSave = b'Brsv'  
Builtin = b'Bltn'  
BurnInH = b'BrnH'  
BurnInM = b'BrnM'  
BurnInS = b'BrnS'  
ButtonMode = b'BtnM'  
CIERGB = b'CRGB'  
CMYK = b'CMYK'  
CMYK64 = b'CMSF'  
CMYKColor = b'ECMY'  
Calculations = b'Clcl'  
Cascade = b'Cscd'  
Center = b'CntR'  
CenterGlow = b'SrcC'  
CenteredFrame = b'CtrF'  
ChannelOptions = b'ChnO'  
ChannelsPaletteOptions = b'ChnP'  
CheckerboardLarge = b'ChcL'  
CheckerboardMedium = b'ChcM'  
CheckerboardNone = b'ChcN'  
CheckerboardSmall = b'ChcS'  
Clear = b'Clar'  
ClearGuides = b'ClrG'  
Clipboard = b'Clpb'  
ClippingPath = b'ClpP'  
CloseAll = b'ClsA'  
CoarseDots = b'CrSD'  
Color = b'Clr '  
ColorBurn = b'CBrn'  
ColorDodge = b'CDdg'
```

```
ColorMatch = b'ClMt'  
ColorNoise = b'ClNs'  
Colorimetric = b'Clrm'  
Composite = b'Cmps'  
ContourCustom = b'sp06'  
ContourDouble = b'sp04'  
ContourGaussian = b'sp02'  
ContourLinear = b'sp01'  
ContourSingle = b'sp03'  
ContourTriple = b'sp05'  
ConvertToCMYK = b'CnvC'  
ConvertToGray = b'CnvG'  
ConvertToLab = b'CnvL'  
ConvertToRGB = b'CnvR'  
CreateDuplicate = b'CrtD'  
CreateInterpolation = b'CrtI'  
Cross = b'Crs '  
CurrentLayer = b'CrrL'  
Custom = b'Cst '  
CustomPattern = b'Cstm'  
CustomStops = b'CstS'  
Cyan = b'Cyn '  
Cyans = b'Cyns'  
Dark = b'Drk '  
Darken = b'Drkn'  
DarkenOnly = b'DrkO'  
DashedLines = b'DshL'  
Desaturate = b'Dstt'  
Diamond = b'Dmnd'  
Difference = b'Dfrn'  
Diffusion = b'Dfsn'  
DiffusionDither = b'DfnD'  
DisplayCursorsPreferences = b'DspC'  
Dissolve = b'Dslv'  
Distort = b'Dstr'  
DodgeH = b'DdgH'
```

```
DodgeM = b'DdgM'  
DodgeS = b'DdgS'  
Dots = b'Dts '  
Draft = b'Drft '  
Duotone = b'Dtn '  
EBUITU = b'EBT '  
EdgeGlow = b'SrcE'  
EliminateEvenFields = b'ElmE'  
EliminateOddFields = b'ElmO'  
Ellipse = b'Elps'  
Emboss = b'Embs '  
Exact = b'Exct '  
Exclusion = b'Xclu'  
FPXCompressLossyJPEG = b'FxJP '  
FPXCompressNone = b'FxNo '  
Faster = b'Dthf '  
File = b'Fle '  
FileInfo = b'FlIn '  
FillBack = b'FlBc '  
FillFore = b'FlFr '  
FillInverse = b'FlIn '  
FillSame = b'FlSm '  
FineDots = b'FnDt '  
First = b'Frst '  
FirstIdle = b'FrId '  
FitOnScreen = b'FtOn '  
ForegroundColor = b'FrgC '  
Forward = b'Frwr '  
FreeTransform = b'FrTr '  
Front = b'Frnt '  
FullDocument = b'Flld '  
FullSize = b'Flsz '  
GIFColorFileColorTable = b'GFCT '  
GIFColorFileColors = b'GFCF '  
GIFColorFileMicrosoftPalette = b'GFMS '  
GIFPaletteAdaptive = b'GFPA '
```

```
GIFPaletteExact = b'GFPE'  
GIFPaletteOther = b'GFPO'  
GIFPaletteSystem = b'GFPS'  
GIFRequiredColorSpaceIndexed = b'GFCI'  
GIFRequiredColorSpaceRGB = b'GFRG'  
GIFRowOrderInterlaced = b'GFIN'  
GIFRowOrderNormal = b'GFNI'  
GaussianDistribution = b'Gsn '  
GeneralPreferences = b'GnrP'  
Good = b'Gd '  
GradientFill = b'GrFl'  
GrainClumped = b'GrnC'  
GrainContrasty = b'GrCn'  
GrainEnlarged = b'GrnE'  
GrainHorizontal = b'GrnH'  
GrainRegular = b'GrnR'  
GrainSoft = b'GrSf'  
GrainSpeckle = b'GrSp'  
GrainSprinkles = b'GrSr'  
GrainStippled = b'GrSt'  
GrainVertical = b'GrnV'  
GrainyDots = b'GrnD'  
Graphics = b'Grp '  
Gray = b'Gry '  
Gray16 = b'GryX'  
Gray18 = b'Gr18'  
Gray22 = b'Gr22'  
Gray50 = b'Gr50'  
GrayScale = b'Gryc'  
Grayscale = b'Grys'  
Green = b'Grn '  
Greens = b'Grns'  
GuidesGridPreferences = b'GudG'  
HDTV = b'HDTV'  
HSBColor = b'HSB1'  
HSLColor = b'HSLC'
```



```
HalftoneFile = b'Hlff'  
HalftoneScreen = b'Hlfs'  
HardLight = b'HrdL'  
Heavy = b'Hvy '  
HideAll = b'HdAl'  
HideSelection = b'HdSl'  
High = b'High'  
HighQuality = b'Hgh '  
Highlights = b'Hghl'  
Histogram = b'Hstg'  
History = b'Hsty'  
HistoryPaletteOptions = b'HstO'  
HistoryPreferences = b'HstP'  
Horizontal = b'Hrzn'  
HorizontalOnly = b'HrzO'  
Hue = b'H '  
IBMPc = b'IBMP'  
ICC = b'ICC '  
Icon = b'Icn '  
IdleVM = b'IdVM'  
Ignore = b'Ignr'  
Image = b'Img '  
ImageCachePreferences = b'ImgP'  
IndexedColor = b'Indl'  
InfoPaletteOptions = b'InfP'  
InfoPaletteToggleSamplers = b'InfT'  
InnerBevel = b'InrB'  
InsetFrame = b'InsF'  
Inside = b'Insd'  
JPEG = b'JPEG'  
JustifyAll = b'JstA'  
JustifyFull = b'JstF'  
KeepProfile = b'KPro'  
KeyboardPreferences = b'KybP'  
Lab = b'Lab '  
Lab48 = b'LbCF'
```

```
LabColor = b'LbCl'  
Large = b'Lrg '  
Last = b'Lst '  
LastFilter = b'LstF'  
LayerOptions = b'LyrO'  
LayersPaletteOptions = b'LyrP'  
Left = b'Left '  
Left_PLUGIN = b'Lft '  
LevelBased = b'LvlB'  
Light = b'Lgt '  
LightBlue = b'LgtB'  
LightDirBottom = b'LDBt'  
LightDirBottomLeft = b'LDBL'  
LightDirBottomRight = b'LDBR'  
LightDirLeft = b'LDLf'  
LightDirRight = b'LDRg'  
LightDirTop = b'LDTp'  
LightDirTopLeft = b'LDTL'  
LightDirTopRight = b'LDTR'  
LightDirectional = b'LghD'  
LightGray = b'LgtG'  
LightOmni = b'LghO'  
LightPosBottom = b'LPBt'  
LightPosBottomLeft = b'LPBL'  
LightPosBottomRight = b'LPBr'  
LightPosLeft = b'LPLf'  
LightPosRight = b'LPRg'  
LightPosTop = b'LPTp'  
LightPosTopLeft = b'LPTL'  
LightPosTopRight = b'LPTR'  
LightRed = b'LgtR'  
LightSpot = b'LghS'  
Lighten = b'Lghn'  
LightenOnly = b'LghO'  
Lightness = b'Lght '  
Line = b'Ln '
```

```
Linear = b'Lnr '  
Lines = b'Lns '  
Linked = b'Lnkd'  
LongLines = b'LngL'  
LongStrokes = b'LngS'  
Low = b'Low '  
LowQuality = b'Lw '  
Lower = b'Lwr '  
Luminosity = b'lmns'  
MacThumbnail = b'McTh'  
Macintosh = b'Mcnt'  
MacintoshSystem = b'McnS'  
Magenta = b'Mgnt'  
Magentas = b'Mgnt'  
Mask = b'Msk '  
MaskedAreas = b'MskA'  
MasterAdaptive = b'MAdp'  
MasterPerceptual = b'MPer'  
MasterSelective = b'MSel'  
Maximum = b'Mxmm'  
MaximumQuality = b'Mxm '  
Maya = b'Maya'  
Medium = b'Mdim'  
MediumBlue = b'MdmB'  
MediumDots = b'MdmD'  
MediumLines = b'MdmL'  
MediumQuality = b'Mdm '  
MediumStrokes = b'MdmS'  
MemoryPreferences = b'MmrP'  
MergeChannels = b'MrgC'  
Merged = b'Mrgd'  
MergedLayers = b'Mrg2'  
MergedLayersOld = b'MrgL'  
Middle = b'Mddl'  
Midtones = b'Mdtn'  
ModeGray = b'MdGr'
```

```
ModeRGB = b'MdRG'  
Monitor = b'Moni'  
MonitorSetup = b'MntS'  
Monotone = b'Mntn'  
Multi72Color = b'72CM'  
Multi72Gray = b'72GM'  
MultiNoCompositePS = b'NCmM'  
Multichannel = b'Mlth'  
Multiply = b'Mltp'  
NTSC = b'NTSC'  
NavigatorPaletteOptions = b'NvgP'  
NearestNeighbor = b'Nrst'  
NetscapeGray = b'NsGr'  
Neutrals = b'Ntrl'  
NewView = b'NwVw'  
Next = b'Nxt '  
Nikon = b'Nkn '  
Nikon105 = b'Nkn1'  
No = b'N '  
NoCompositePS = b'NCmp'  
Normal = b'Nrml'  
NormalPath = b'NrmP'  
Null = b'null'  
OS2 = b'OS2 '  
Off = b'Off '  
On = b'On '  
OpenAs = b'OpAs'  
Orange = b'Orng'  
OutFromCenter = b'OtFr'  
OutOfGamut = b'OtOf'  
OuterBevel = b'OtrB'  
OutsetFrame = b'OutF'  
Outside = b'Otsd'  
Overlay = b'Ovrl'  
P22EBU = b'P22B'  
PNGFilterAdaptive = b'PGAd'
```

```
PNGFilterAverage = b'PGAv'  
PNGFilterNone = b'PGNo'  
PNGFilterPaeth = b'PGPt'  
PNGFilterSub = b'PGSb'  
PNGFilterUp = b'PGUp'  
PNGInterlaceAdam7 = b'PGIA'  
PNGInterlaceNone = b'PGIN'  
PagePosCentered = b'PgPC'  
PagePosTopLeft = b'PgTL'  
PageSetup = b'PgSt'  
PaintbrushEraser = b'Pntb'  
PalSecam = b'PlSc'  
PanaVision = b'PnVs'  
PathsPaletteOptions = b'PthP'  
Pattern = b'Ptrn'  
PatternDither = b'PtnD'  
PencilEraser = b'Pncl'  
Perceptual = b'Perc'  
Perspective = b'Prsp'  
PhotoshopPicker = b'Phtk'  
PickCMYK = b'PckC'  
PickGray = b'PckG'  
PickHSB = b'PckH'  
PickLab = b'PckL'  
PickOptions = b'PckO'  
PickRGB = b'PckR'  
PillowEmboss = b'PlEb'  
PixelPaintSize1 = b'PxS1'  
PixelPaintSize2 = b'PxS2'  
PixelPaintSize3 = b'PxS3'  
PixelPaintSize4 = b'PxS4'  
Place = b'Plce'  
PlaybackOptions = b'PbkO'  
PluginPicker = b'PlgP'  
PluginsScratchDiskPreferences = b'PlgS'  
PolarToRect = b'PlrR'
```

```
PondRipples = b'PndR'  
Precise = b'Prc '  
PreciseMatte = b'PrBL'  
PreviewBlack = b'PrvB'  
PreviewCMY = b'PrvN'  
PreviewCMYK = b'PrvC'  
PreviewCyan = b'Prvy'  
PreviewMagenta = b'PrvM'  
PreviewOff = b'PrvO'  
PreviewYellow = b'PrvY'  
Previous = b'Prvs'  
Primaries = b'Prim'  
PrintSize = b'PrnS'  
PrintingInksSetup = b'PrnI'  
Purple = b'Prp '  
Pyramids = b'Pyrm'  
QCSAverage = b'Qcsa'  
QCSCorner0 = b'Qcs0'  
QCSCorner1 = b'Qcs1'  
QCSCorner2 = b'Qcs2'  
QCSCorner3 = b'Qcs3'  
QCSIndependent = b'Qcsi'  
QCSSide0 = b'Qcs4'  
QCSSide1 = b'Qcs5'  
QCSSide2 = b'Qcs6'  
QCSSide3 = b'Qcs7'  
Quadtone = b'Qdtn'  
QueryAlways = b'QurA'  
QueryAsk = b'Qurl'  
QueryNever = b'QurN'  
RGB = b'RGB '  
RGB48 = b'RGBF'  
RGBColor = b'RGBC'  
Radial = b'Rdl '  
Random = b'Rndm'  
RectToPolar = b'RctP'
```

```
Red = b'Rd '
RedrawComplete = b'RdCm'
Reds = b'Rds '
Reflected = b'Rflc'
Relative = b'Rltv'
Repeat = b'Rpt '
RepeatEdgePixels = b'RptE'
RevealAll = b'Rv1A'
RevealSelection = b'Rv1S'
Revert = b'Rvrt'
Right = b'Rght'
Rotate = b'Rtte'
RotoscopingPreferences = b'RtsP'
Round = b'Rnd '
RulerCm = b'RrCm'
RulerInches = b'RrIn'
RulerPercent = b'RrPr'
RulerPicas = b'RrPi'
RulerPixels = b'RrPx'
RulerPoints = b'RrPt'
SMPTEC = b'SMPC'
SRGB = b'SRGB'
Sample3x3 = b'Smp3'
Sample5x5 = b'Smp5'
SamplePoint = b'SmpP'
Saturate = b'Str '
Saturation = b'Strt'
SaveForWeb = b'Svfw'
Saved = b'Sved'
SavingFilesPreferences = b'SvnF'
Scale = b'Scl '
Screen = b'Scrn'
ScreenCircle = b'ScrC'
ScreenDot = b'ScrD'
ScreenLine = b'ScrL'
SelectedAreas = b'Slca'
```

Selection = b'Slct'  
Selective = b'Sele'  
SeparationSetup = b'SprS'  
SeparationTables = b'SprT'  
Shadows = b'Shdw'  
ShortLines = b'ShrL'  
ShortStrokes = b'ShSt'  
Single72Color = b'72CS'  
Single72Gray = b'72GS'  
SingleNoCompositePS = b'NCmS'  
Skew = b'Skew'  
SlopeLimitMatte = b'Slmt'  
Small = b'Sml '  
SmartBlurModeEdgeOnly = b'SBME'  
SmartBlurModeNormal = b'SBMN'  
SmartBlurModeOverlayEdge = b'SBMO'  
SmartBlurQualityHigh = b'SBQH'  
SmartBlurQualityLow = b'SBQL'  
SmartBlurQualityMedium = b'SBQM'  
Snapshot = b'Snps'  
SoftLight = b'SftL'  
SoftMatte = b'SfBL'  
SolidColor = b'SClr'  
Spectrum = b'Spct'  
Spin = b'Spn '  
SpotColor = b'Spot'  
Square = b'Sqr '  
Stagger = b'Stgr'  
StampIn = b'In '  
StampOut = b'Out '  
Standard = b'Std '  
StdA = b'StdA'  
StdB = b'StdB'  
StdC = b'StdC'  
StdE = b'StdE'  
StretchToFit = b'StrF'



```
StrokeDirHorizontal = b'SDHZ'  
StrokeDirLeftDiag = b'SDLD'  
StrokeDirRightDiag = b'SDRD'  
StrokeDirVertical = b'SDVT'  
StylesAppend = b'SLSA'  
StylesDelete = b'SLSF'  
StylesLoad = b'SLSD'  
StylesNew = b'SLSN'  
StylesReset = b'SLSR'  
StylesSave = b'SLSV'  
Subtract = b'SBTR'  
SwatchesAppend = b'SWTA'  
SwatchesReplace = b'SWTP'  
SwatchesReset = b'SWTR'  
SwatchesSave = b'SWTS'  
SystemPicker = b'SYSP'  
TIFF = b'TIFF'  
Tables = b'Tbl '  
Target = b'Trgt'  
TargetPath = b'Trgp'  
TexTypeBlocks = b'TxBL'  
TexTypeBrick = b'TXBR'  
TexTypeBurlap = b'TXBU'  
TexTypeCanvas = b'TXCA'  
TexTypeFrosted = b'TXFR'  
TexTypeSandstone = b'TXST'  
TexTypeTinyLens = b'TXTL'  
Threshold = b'THRH'  
Thumbnail = b'THMB'  
Tile = b'TILE'  
Tile_PLUGIN = b'Tl '  
ToggleActionsPalette = b'TgLA'  
ToggleBlackPreview = b'TgBP'  
ToggleBrushesPalette = b'TglB'  
ToggleCMYKPreview = b'TglC'  
ToggleCMYPreview = b'TgCM'
```

```
ToggleChannelsPalette = b'Tglh'  
ToggleColorPalette = b'Tglc'  
ToggleCyanPreview = b'TgCP'  
ToggleDocumentPalette = b'TgDc'  
ToggleEdges = b'TglE'  
ToggleGamutWarning = b'TglG'  
ToggleGrid = b'TgGr'  
ToggleGuides = b'Tgld'  
ToggleHistoryPalette = b'TglH'  
ToggleInfoPalette = b'TglI'  
ToggleLayerMask = b'TglM'  
ToggleLayersPalette = b'Tgly'  
ToggleLockGuides = b'TglL'  
ToggleMagentaPreview = b'TgMP'  
ToggleNavigatorPalette = b'TglN'  
ToggleOptionsPalette = b'TglO'  
TogglePaths = b'TglP'  
TogglePathsPalette = b'Tglt'  
ToggleRGBMacPreview = b'TrMp'  
ToggleRGBUncompensatedPreview = b'TrUp'  
ToggleRGBWindowsPreview = b'TrWp'  
ToggleRulers = b'TglR'  
ToggleSnapToGrid = b'TgSn'  
ToggleSnapToGuides = b'TglS'  
ToggleStatusBar = b'Tgls'  
ToggleStylesPalette = b'TgSl'  
ToggleSwatchesPalette = b'Tglw'  
ToggleToolsPalette = b'TglT'  
ToggleYellowPreview = b'TgYP'  
Top = b'Top '  
Transparency = b'Trsp'  
TransparencyGamutPreferences = b'TrnG'  
Transparent = b'Trns'  
Trinitron = b'Trnt'  
Tritone = b'Trtn'  
UIBitmap = b'UBtm'
```

```
UICMYK = b'UCMY'  
UIDuotone = b'UDtn'  
UIGrayscale = b'UGry'  
UIIndexed = b'UInd'  
UILab = b'ULab'  
UIMultichannel = b'UMlt'  
UIRGB = b'URGB'  
Undo = b'Und '  
Uniform = b'Unfm'  
UniformDistribution = b'Unfr'  
UnitsRulersPreferences = b'UntrR'  
Upper = b'Upr '  
UserStop = b'UsrS'  
VMPreferences = b'VMPr'  
Vertical = b'Vrtc'  
VerticalOnly = b'VrtO'  
Violet = b'Vlt '  
WaveSine = b'WvSn'  
WaveSquare = b'WvSq'  
WaveTriangle = b'WvTr'  
Web = b'Web '  
White = b'Wht '  
Whites = b'Whts'  
WideGamutRGB = b'WRGB'  
WidePhosphors = b'Wide'  
WinThumbnail = b'WnTh'  
Wind = b'Wnd '  
Windows = b'Win '  
WindowsSystem = b'WndS'  
WorkPath = b'WrkP'  
Wrap = b'Wrp '  
WrapAround = b'WrpA'  
Yellow = b'Yllw'  
YellowColor = b'Ylw '  
Yellows = b'Ylws'  
Yes = b'Ys '
```

```
Zip = b'ZpEn'
Zoom = b'Zm '
ZoomIn = b'ZmIn'
ZoomOut = b'ZmOt'
_16BitsPerPixel = b'16Bt'
_1BitPerPixel = b'OnBt'
_2BitsPerPixel = b'2Bts'
_32BitsPerPixel = b'32Bt'
_4BitsPerPixel = b'4Bts'
_5000 = b'5000'
_5500 = b'5500'
_6500 = b'6500'
_72Color = b'72C1'
_72Gray = b'72Gr'
_7500 = b'7500'
_8BitsPerPixel = b'EghB'
_9300 = b'9300'
_None = b'None'
_generate_next_value_ (start, count, last_values)
_member_map_ = {'A': <Enum.A: b'A '>, 'ADSBottoms': <Enum.ADSBottoms: b'AdBt'>, 'ADS'
_member_names_ = ['Add', 'AmountHigh', 'AmountLow', 'AmountMedium', 'AntiAliasNone', 'A'
_member_type_
    alias of builtins.bytes
_value2member_map_ = {b'1565': <Enum.BitDepthA1R5G5B5: b'1565'>, b'16Bt': <Enum._16
```

### 3.24.3 Event

**class** psd\_tools.terminology.Event

Event definitions extracted from PITerminology.h.

See <https://www.adobe.com/devnet/photoshop/sdk.html>

```
AccentedEdges = b'AccE'
```

```
Add = b'Add '
```

```
AddNoise = b'AdNs'
```

```
AddTo = b'AddT'
```

```
Align = b'Algn'
```

```
All = b'All '
```

```
AngledStrokes = b'AngS'
```

```
ApplyImage = b'AppI'
```

ApplyStyle = b'ASty'  
Assert = b'Asrt'  
Average = b'Avrg'  
BackLight = b'BacL'  
BasRelief = b'BsRl'  
Batch = b'Btch'  
BatchFromDroplet = b'BtcF'  
Blur = b'Blr '  
BlurMore = b'BlrM'  
Border = b'Brdr'  
Brightness = b'BrgC'  
CanvasSize = b'CnvS'  
ChalkCharcoal = b'ChlC'  
ChannelMixer = b'ChnM'  
Charcoal = b'Chrc'  
Chrome = b'Chrm'  
Clear = b'Cler'  
Close = b'Cls '  
Clouds = b'Clds'  
ColorBalance = b'ClrB'  
ColorCast = b'ColE'  
ColorHalftone = b'ClrH'  
ColorRange = b'ClrR'  
ColoredPencil = b'ClrP'  
ConteCrayon = b'CntC'  
Contract = b'Cntc'  
ConvertMode = b'CnvM'  
Copy = b'copy'  
CopyEffects = b'CpFX'  
CopyMerged = b'CpyM'  
CopyToLayer = b'CpTL'  
Craquelure = b'Crql'  
CreateDroplet = b'CrtD'  
Crop = b'Crop'  
Crosshatch = b'Crsh'  
Crystallize = b'Crst'

Curves = b'Crvs'  
Custom = b'Cstm'  
Cut = b'cut '  
CutToLayer = b'CtTL'  
Cutout = b'Ct '  
DarkStrokes = b'DrkS'  
DeInterlace = b'Dntr'  
DefinePattern = b'DfnP'  
Defringe = b'Dfrg'  
Delete = b'Dlt '  
Desaturate = b'Dstt'  
Deselect = b'Dslc'  
Despeckle = b'Dspc'  
DifferenceClouds = b'DfrC'  
Diffuse = b'Dfs '  
DiffuseGlow = b'DfsG'  
DisableLayerFX = b'dlfx'  
Displace = b'Dspl'  
Distribute = b'Dstr'  
Draw = b'Draw'  
DryBrush = b'DryB'  
Duplicate = b'Dplc'  
DustAndScratches = b'DstS'  
Emboss = b'Embs'  
Equalize = b'Eqlz'  
Exchange = b'Exch'  
Expand = b'Expn'  
Export = b'Expr'  
Extrude = b'Extr'  
Facet = b'Fct '  
Fade = b'Fade'  
Feather = b'Fthr'  
Fibers = b'Fbrs'  
Fill = b'Fl '  
FilmGrain = b'FlmG'  
Filter = b'Fltr'

```
FindEdges = b'FndE'  
FlattenImage = b'FltI'  
Flip = b'Flip'  
Fragment = b'Frgm'  
Fresco = b'Frsc'  
GaussianBlur = b'GsnB'  
Get = b'getd'  
Glass = b'Gls '  
GlowingEdges = b'GlwE'  
Gradient = b'Grdn'  
GradientMap = b'GrMp'  
Grain = b'Grn '  
GraphicPen = b'GraP'  
Group = b'GrpL'  
Grow = b'Grow'  
HSBHSL = b'HsbP'  
HalftoneScreen = b'Hlfs'  
Hide = b'Hd '  
HighPass = b'HghP'  
HueSaturation = b'HStr'  
ImageSize = b'ImgS'  
Import = b'Impr'  
InkOutlines = b'InkO'  
Intersect = b'Intr'  
IntersectWith = b'IntW'  
Inverse = b'Invs'  
Invert = b'Invr'  
LensFlare = b'LnsF'  
Levels = b'Lvls'  
LightingEffects = b'LghE'  
Link = b'Lnk '  
Make = b'Mk '  
Maximum = b'Mxm '  
Median = b'Mdn '  
MergeLayers = b'Mrg2'  
MergeLayersOld = b'MrgL'
```

MergeSpotChannel = b'MSpt'  
MergeVisible = b'MrgV'  
Mezzotint = b'Mztn'  
Minimum = b'Mnm '  
Mosaic = b'Msc '  
Mosaic\_PLUGIN = b'MscT'  
MotionBlur = b'MtnB'  
Move = b'move'  
NTSCColors = b'NTSC'  
NeonGlow = b'NGlw'  
Next = b'Nxt '  
NotePaper = b'NtPr'  
Notify = b'Ntfy'  
Null = b'null'  
OceanRipple = b'OcnR'  
Offset = b'Ofst'  
Open = b'Opn '  
OpenUntitled = b'OpnU'  
PaintDaubs = b'PntD'  
PaletteKnife = b'PltK'  
Paste = b'past'  
PasteEffects = b'PaFX'  
PasteInto = b'PstI'  
PasteOutside = b'PstO'  
Patchwork = b'Ptch'  
Photocopy = b'Phtc'  
Pinch = b'Pnch'  
Place = b'Plc '  
Plaster = b'Plst'  
PlasticWrap = b'PlsW'  
Play = b'Ply '  
Pointillize = b'Pntl'  
Polar = b'Plr '  
PosterEdges = b'PstE'  
Posterize = b'Pstr'  
Previous = b'Prvs'



```
Print = b'Prnt'  
ProfileToProfile = b'PrfT'  
Purge = b'Prge'  
Quit = b'quit'  
RadialBlur = b'RdlB'  
Rasterize = b'Rstr'  
RasterizeTypeSheet = b'RstT'  
RemoveBlackMatte = b'RmvB'  
RemoveLayerMask = b'RmvL'  
RemoveWhiteMatte = b'RmvW'  
Rename = b'Rnm '  
ReplaceColor = b'RplC'  
Reset = b'Rset'  
Reticulation = b'Rtcl'  
Revert = b'Rvrt'  
Ripple = b'Rple'  
Rotate = b'Rtte'  
RoughPastels = b'RghP'  
Save = b'save'  
Select = b'slct'  
SelectiveColor = b'SlcC'  
Set = b'setd'  
Sharpen = b'Shrp'  
SharpenEdges = b'ShrE'  
SharpenMore = b'ShrM'  
Shear = b'Shr '  
Show = b'Shw '  
Similar = b'Smlr'  
SmartBlur = b'SmrB'  
Smooth = b'Smth'  
SmudgeStick = b'SmdS'  
Solarize = b'Slrz'  
Spatter = b'Spt '  
Spherize = b'Sphr'  
SplitChannels = b'SplC'  
Sponge = b'Spng'
```

```
SprayedStrokes = b'SprS'  
StainedGlass = b'StnG'  
Stamp = b'Stmp'  
Stop = b'Stop'  
Stroke = b'Strk'  
Subtract = b'Sbtr'  
SubtractFrom = b'SbtF'  
Sumie = b'Smie'  
TakeMergedSnapshot = b'TkMr'  
TakeSnapshot = b'TkSn'  
TextureFill = b'TxtF'  
Texturizer = b'Txtz'  
Threshold = b'Thrs'  
Tiles = b'Tls '  
TornEdges = b'TrnE'  
TraceContour = b'TrcC'  
Transform = b'Trnf'  
Trap = b'Trap'  
Twirl = b'Twrl'  
Underpainting = b'Undr'  
Undo = b'undo'  
Ungroup = b'Ungr'  
Unlink = b'Unlk'  
UnsharpMask = b'UnsM'  
Variations = b'Vrtn'  
Wait = b'Wait'  
WaterPaper = b'WtrP'  
Watercolor = b'Wtrc'  
Wave = b'Wave'  
Wind = b'Wnd '  
ZigZag = b'ZgZg'  
_3DTransform = b'TdT '  
_generate_next_value_(start, count, last_values)  
_member_map_ = {'AccentedEdges': <Event.AccentedEdges: b'AcceE'>, 'Add': <Event.Add:  
_member_names_ = ['_3DTransform', 'Average', 'ApplyStyle', 'Assert', 'AccentedEdges',
```

```

_member_type_
    alias of builtins.bytes

_value2member_map_ = {b'ASty': <Event.ApplyStyle: b'ASty'>, b'AccE': <Event.Accented

```

### 3.24.4 Form

**class** psd\_tools.terminology.**Form**

Form definitions extracted from PITerminology.h.

See <https://www.adobe.com/devnet/photoshop/sdk.html>

```

Class = b'Class'
Enumerated = b'Enmr'
Identifier = b'Idnt'
Index = b'indx'
Offset = b'rele'
Property = b'prop'
_generate_next_value_(start, count, last_values)
_member_map_ = {'Class': <Form.Class: b'Class'>, 'Enumerated': <Form.Enumerated: b'
_member_names_ = ['Class', 'Enumerated', 'Identifier', 'Index', 'Offset', 'Property']
_member_type_
    alias of builtins.bytes

_value2member_map_ = {b'Class': <Form.Class: b'Class'>, b'Enmr': <Form.Enumerated: b

```

### 3.24.5 Key

**class** psd\_tools.terminology.**Key**

Key definitions extracted from PITerminology.h.

See <https://www.adobe.com/devnet/photoshop/sdk.html>

```

A = b'A '
Adjustment = b'Adjs'
Aligned = b'Algd'
Alignment = b'Algn'
AllExcept = b'AllE'
AllPS = b'All '
AllToolOptions = b'AlTl'
AlphaChannelOptions = b'AChn'
AlphaChannels = b'AlpC'
AmbientBrightness = b'AmbB'
AmbientColor = b'AmbC'
Amount = b'Amnt'

```

AmplitudeMax = b'AmMx'  
AmplitudeMin = b'AmMn'  
Anchor = b'Anch'  
Angle = b'Angl'  
Angle1 = b'Ang1'  
Angle2 = b'Ang2'  
Angle3 = b'Ang3'  
Angle4 = b'Ang4'  
AntiAlias = b'AntA'  
Append = b'Appe'  
Apply = b'Aply'  
Area = b'Ar '  
Arrowhead = b'Arrw'  
As = b'As '  
AssetBin = b'Asst'  
AssumedCMYK = b'AssC'  
AssumedGray = b'AssG'  
AssumedRGB = b'AssR'  
At = b'At '  
Auto = b'Auto'  
AutoContrast = b'AuCo'  
AutoErase = b'Atrs'  
AutoKern = b'AtKr'  
AutoUpdate = b'AtUp'  
Axis = b'Axis'  
B = b'B '  
Background = b'Bckg'  
BackgroundColor = b'BckC'  
BackgroundLevel = b'BckL'  
Backward = b'Bwd '  
Balance = b'Blnc'  
BaselineShift = b'Bsln'  
BeepWhenDone = b'BpWh'  
BeginRamp = b'BgnR'  
BeginSustain = b'BgnS'  
BevelDirection = b'bvlD'

```
BevelEmboss = b'ebbl'  
BevelStyle = b'bvls'  
BevelTechnique = b'bvlt'  
BigNudgeH = b'BgNH'  
BigNudgeV = b'BgNV'  
BitDepth = b'BtDp'  
Black = b'Blck'  
BlackClip = b'BlcC'  
BlackGeneration = b'Blcn'  
BlackGenerationCurve = b'BlcG'  
BlackIntensity = b'BlcI'  
BlackLevel = b'BlcL'  
BlackLimit = b'BlcL'  
Bleed = b'Bld '  
BlendRange = b'Blnd'  
Blue = b'Bl '  
BlueBlackPoint = b'BlBl'  
BlueGamma = b'BlGm'  
BlueWhitePoint = b'BlWh'  
BlueX = b'BlX '  
BlueY = b'BlY '  
Blur = b'blur'  
BlurMethod = b'BlrM'  
BlurQuality = b'BlrQ'  
Book = b'Bk '  
BorderThickness = b'BrdT'  
Bottom = b'Btom'  
Brightness = b'Brgh'  
BrushDetail = b'BrsD'  
BrushSize = b'BrsS'  
BrushType = b'BrsT'  
Brushes = b'Brsh'  
BumpAmplitude = b'BmpA'  
BumpChannel = b'BmpC'  
By = b'By '  
Byline = b'Byln'
```

BylineTitle = b'BylT'  
ByteOrder = b'BytO'  
CMYKSetup = b'CMYS'  
CachePrefs = b'CchP'  
Calculation = b'Clcl'  
CalibrationBars = b'Clbr'  
Caption = b'Cptn'  
CaptionWriter = b'CptW'  
Category = b'Ctgr'  
CellSize = b'ClSz'  
Center = b'CntR'  
CenterCropMarks = b'CntC'  
ChalkArea = b'ChlA'  
Channel = b'Chnl'  
ChannelMatrix = b'ChMx'  
ChannelName = b'ChnN'  
Channels = b'Chns'  
ChannelsInterleaved = b'ChnI'  
CharcoalAmount = b'ChAm'  
CharcoalArea = b'ChrA'  
ChokeMatte = b'Ckmt'  
ChromeFX = b'ChFX'  
City = b'City'  
ClearAmount = b'ClrA'  
ClippingPath = b'ClPt'  
ClippingPathEPS = b'ClpP'  
ClippingPathFlatness = b'ClpF'  
ClippingPathIndex = b'ClpI'  
ClippingPathInfo = b'Clpg'  
CloneSource = b'ClnS'  
ClosedSubpath = b'Clsp'  
Color = b'Clr '  
ColorChannels = b'Clrh'  
ColorCorrection = b'ClrC'  
ColorIndicates = b'ClrI'  
ColorManagement = b'ClMg'

---

```
ColorPickerPrefs = b'Clrr'  
ColorSpace = b'ClrS'  
ColorTable = b'ClrT'  
Colorize = b'Clrz'  
Colors = b'Clrs'  
ColorsList = b'ClrL'  
ColumnWidth = b'ClmW'  
CommandKey = b'CmdK'  
Compensation = b'Cmpn'  
Compression = b'Cmpr'  
Concavity = b'Cncv'  
Condition = b'Cndt'  
Constant = b'Cnst'  
Constrain = b'Cnst'  
ConstrainProportions = b'CnsP'  
ConstructionFOV = b'Cfov'  
Contiguous = b'Cntg'  
Continue = b'Cntn'  
Continuity = b'Cnty'  
ContourType = b'ShpC'  
Contrast = b'Cntr'  
Convert = b'Cnvr'  
Copy = b'Cpy '  
Copyright = b'Cpyr'  
CopyrightNotice = b'CprN'  
CornerCropMarks = b'CrnC'  
Count = b'Cnt '  
CountryName = b'CntN'  
CrackBrightness = b'CrcB'  
CrackDepth = b'CrcD'  
CrackSpacing = b'CrcS'  
CreateLayersFromLayerFX = b'blfl'  
Credit = b'Crdt'  
Crossover = b'Crss'  
Current = b'Crnt'  
CurrentHistoryState = b'CrnH'
```

CurrentLight = b'CrnL'  
CurrentToolOptions = b'CrnT'  
Curve = b'Crv '  
CurveFile = b'CrvF'  
Custom = b'Cstm'  
CustomForced = b'CstF'  
CustomMatte = b'CstM'  
CustomPalette = b'CstP'  
Cyan = b'Cyn '  
DCS = b'DCS '  
DPXFormat = b'DPXf'  
DarkIntensity = b'DrkI'  
Darkness = b'Drkn'  
DateCreated = b'DtCr'  
Datum = b'Dt '  
Definition = b'Dfnt'  
Density = b'Dnst'  
Depth = b'Dpth'  
DestBlackMax = b'Dstl'  
DestBlackMin = b'DstB'  
DestWhiteMax = b'Dstt'  
DestWhiteMin = b'DstW'  
DestinationMode = b'DstM'  
Detail = b'Dtl '  
Diameter = b'Dmtr'  
DiffusionDither = b'Dffd'  
Direction = b'Drct'  
DirectionBalance = b'DrcB'  
DisplaceFile = b'DspF'  
DisplacementMap = b'DspM'  
DisplayPrefs = b'DspP'  
Distance = b'Dstn'  
Distortion = b'Dstr'  
Distribution = b'Dstr'  
Dither = b'Dthr'  
DitherAmount = b'DthA'



DitherPreserve = b'Dthp'  
DitherQuality = b'Dthq'  
DocumentID = b'DocI'  
DotGain = b'DtGn'  
DotGainCurves = b'DtGC'  
DropShadow = b'DrSh'  
Duplicate = b'Dplc'  
DynamicColorSliders = b'DnmC'  
Edge = b'Edg '  
EdgeBrightness = b'EdgB'  
EdgeFidelity = b'EdgF'  
EdgeIntensity = b'EdgI'  
EdgeSimplicity = b'EdgS'  
EdgeThickness = b'EdgT'  
EdgeWidth = b'EdgW'  
Effect = b'Effc'  
EmbedCMYK = b'EmbC'  
EmbedGray = b'EmbG'  
EmbedLab = b'EmbL'  
EmbedProfiles = b'EmbP'  
EmbedRGB = b'EmbR'  
EmulsionDown = b'Emld'  
EnableGestures = b'EGst'  
Enabled = b'enab'  
Encoding = b'Encd'  
End = b'End '  
EndArrowhead = b'EndA'  
EndRamp = b'EndR'  
EndSustain = b'EndS'  
Engine = b'Engn'  
EraseToHistory = b'ErsT'  
EraserKind = b'ErsK'  
ExactPoints = b'ExcP'  
Export = b'Expr'  
ExportClipboard = b'ExpC'  
Exposure = b'Exps'

```
Extend = b'Extd'  
ExtendedQuality = b'EQlt'  
Extension = b'Extn'  
ExtensionsQuery = b'ExtQ'  
ExtrudeDepth = b'ExtD'  
ExtrudeMaskIncomplete = b'ExtM'  
ExtrudeRandom = b'ExtR'  
ExtrudeSize = b'ExtS'  
ExtrudeSolidFace = b'ExtF'  
ExtrudeType = b'ExtT'  
EyeDropperSample = b'EyDr'  
FPXCompress = b'FxCm'  
FPXQuality = b'FxQl'  
FPXSize = b'FxCm'  
FPXView = b'FxCm'  
FadeTo = b'FdT '  
FadeoutSteps = b'FdtS'  
Falloff = b'FlOf'  
Feather = b'Fthr'  
FiberLength = b'FbrL'  
File = b'File'  
FileCreator = b'FlCr'  
FileInfo = b'FlIn'  
FileReference = b'FlR'  
FileSavePrefs = b'FlSP'  
FileType = b'FlTy'  
FilesList = b'flst'  
Fill = b'Fl '  
FillColor = b'FlCl'  
FillNeutral = b'FlNt'  
FilterLayerPersistentData = b'FlPd'  
FilterLayerRandomSeed = b'FlRs'  
Fingerpainting = b'Fngr'  
FlareCenter = b'FlrC'  
Flatness = b'Fltn'  
Flatten = b'Fltt'
```

```
FlipVertical = b'FlpV'  
Focus = b'Fcs '  
Folders = b'Fldr'  
FontDesignAxes = b'FntD'  
FontDesignAxesVectors = b'FntV'  
FontName = b'FntN'  
FontScript = b'Scrp'  
FontStyleName = b'FntS'  
FontTechnology = b'FntT'  
ForcedColors = b'FrcC'  
ForegroundColor = b'FrgC'  
ForegroundLevel = b'FrgL'  
Format = b'Fmt '  
Forward = b'Fwd '  
FrameFX = b'FrFX'  
FrameWidth = b'FrmW'  
FreeTransformCenterState = b'FTcs'  
Frequency = b'Frqn'  
From = b'From'  
FromBuiltin = b'FrmB'  
FromMode = b'FrmM'  
FunctionKey = b'FncK'  
Fuzziness = b'Fzns'  
GCR = b'GCR '  
GIFColorFileType = b'GFPT'  
GIFColorLimit = b'GFCL'  
GIFExportCaption = b'GFEC'  
GIFMaskChannelIndex = b'GFMI'  
GIFMaskChannelInverted = b'GFMV'  
GIFPaletteFile = b'GFPF'  
GIFPaletteType = b'GFPL'  
GIFRequiredColorSpaceType = b'GFCS'  
GIFRowOrderType = b'GFIT'  
GIFTransparentColor = b'GFTC'  
GIFTransparentColorBlue = b'GFTB'  
GIFTransparentColorGreen = b'GFTG'
```

```
GIFTransparentIndexRed = b'GFTR'  
GIFUseBestMatch = b'GFBM'  
Gamma = b'Gmm '  
GamutWarning = b'GmtW'  
GeneralPrefs = b'GnrP'  
GlobalAngle = b'gblA'  
GlobalLightingAngle = b'gagl'  
Gloss = b'Glos'  
GlowAmount = b'GlwA'  
GlowTechnique = b'GlwT'  
Gradient = b'Grad'  
GradientFill = b'Grdf'  
Grain = b'Grn '  
GrainType = b'Grnt '  
Graininess = b'Grns '  
Gray = b'Gry '  
GrayBehavior = b'GrBh'  
GraySetup = b'GrSt '  
Green = b'Grn '  
GreenBlackPoint = b'GrnB'  
GreenGamma = b'GrnG'  
GreenWhitePoint = b'GrnW'  
GreenX = b'GrnX'  
GreenY = b'GrnY'  
GridColor = b'GrdC'  
GridCustomColor = b'Grds '  
GridMajor = b'GrdM'  
GridMinor = b'Grdn '  
GridStyle = b'GrdS '  
GridUnits = b'Grdt '  
Group = b'Grup '  
GroutWidth = b'GrtW'  
GrowSelection = b'GrwS '  
Guides = b'Gdes '  
GuidesColor = b'GdsC '  
GuidesCustomColor = b'Gdss '
```

```
GuidesPrefs = b'GdPr'  
GuidesStyle = b'GdsS'  
GutterWidth = b'GttW'  
HalftoneFile = b'Hlff'  
HalftoneScreen = b'Hlfs'  
HalftoneSize = b'HlSz'  
HalftoneSpec = b'Hlfp'  
Hardness = b'Hrdn'  
HasCmdHPreference = b'HCdH'  
Header = b'Hdr '  
Headline = b'Hdln'  
Height = b'Hght'  
HighlightArea = b'HghA'  
HighlightColor = b'hglC'  
HighlightLevels = b'HghL'  
HighlightMode = b'hglM'  
HighlightOpacity = b'hglO'  
HighlightStrength = b'HghS'  
HistoryBrushSource = b'HstB'  
HistoryPrefs = b'HstP'  
HistoryStateSource = b'HsSS'  
HistoryStates = b'HsSt'  
Horizontal = b'Hrzn'  
HorizontalScale = b'HrzS'  
HostName = b'HstN'  
HostVersion = b'HstV'  
Hue = b'H '  
ICCEngine = b'ICCE'  
ICCSetupName = b'ICct'  
ID = b'Idnt'  
Idle = b'Idle'  
ImageBalance = b'ImgB'  
Import = b'Impr'  
Impressionist = b'Imps'  
In = b'In '  
Inherits = b'c@#^'
```

```
InkColors = b'InkC'  
Inks = b'Inks'  
InnerGlow = b'IrGl'  
InnerGlowSource = b'glwS'  
InnerShadow = b'IrSh'  
Input = b'Inpt'  
InputBlackPoint = b'kIBP'  
InputMapRange = b'Inmr'  
InputRange = b'Inpr'  
InputWhitePoint = b'kIWP'  
Intensity = b'Intn'  
Intent = b'Inte'  
InterfaceBevelHighlight = b'IntH'  
InterfaceBevelShadow = b'Intv'  
InterfaceBlack = b'IntB'  
InterfaceBorder = b'Intd'  
InterfaceButtonDarkShadow = b'Intk'  
InterfaceButtonDownFill = b'Intt'  
InterfaceButtonUpFill = b'InBF'  
InterfaceColorBlue2 = b'ICBL'  
InterfaceColorBlue32 = b'ICBH'  
InterfaceColorGreen2 = b'ICGL'  
InterfaceColorGreen32 = b'ICGH'  
InterfaceColorRed2 = b'ICRL'  
InterfaceColorRed32 = b'ICRH'  
InterfaceIconFillActive = b'IntI'  
InterfaceIconFillDimmed = b'IntF'  
InterfaceIconFillSelected = b'Intc'  
InterfaceIconFrameActive = b'Intm'  
InterfaceIconFrameDimmed = b'Intr'  
InterfaceIconFrameSelected = b'IntS'  
InterfacePaletteFill = b'IntP'  
InterfaceRed = b'IntR'  
InterfaceToolTipBackground = b'IntT'  
InterfaceToolTipText = b'ITTT'  
InterfaceTransparencyBackground = b'ITBg'
```

```
InterfaceTransparencyForeground = b'ITFg'  
InterfaceWhite = b'IntW'  
Interlace = b'Intr'  
InterlaceCreateType = b'IntC'  
InterlaceEliminateType = b'IntE'  
Interpolation = b'Intr'  
InterpolationMethod = b'IntM'  
Invert = b'Invr'  
InvertMask = b'InvM'  
InvertSource2 = b'InvS'  
InvertTexture = b'InvT'  
IsDirty = b'IsDr'  
ItemIndex = b'ItmI'  
JPEGQuality = b'JPEQ'  
Kerning = b'Krng'  
Keywords = b'Kywd'  
Kind = b'Knd '  
LUTAnimation = b'LTnm'  
LZWCompression = b'LZWC'  
Labels = b'LbIs'  
Landscape = b'Lnds'  
LastTransform = b'LstT'  
Layer = b'Lyr '  
LayerEffects = b'Lefx'  
LayerFXVisible = b'lfxv'  
LayerID = b'LyrI'  
LayerName = b'LyrN'  
Layers = b'Lyrs'  
Leading = b'Ldng'  
Left = b'Left'  
LegacySerialString = b'lSNs'  
Length = b'Lngt'  
Lens = b'Lns '  
Level = b'Lvl '  
Levels = b'Lvls'  
LightDark = b'LgDr'
```

LightDirection = b'LghD'  
LightIntensity = b'LghI'  
LightPosition = b'LghP'  
LightSource = b'LghS'  
LightType = b'LghT'  
LightenGrout = b'LghG'  
Lightness = b'Lght'  
Line = b'Line'  
LinkEnable = b'lnkE'  
LinkedLayerIDs = b'LnkL'  
LocalLightingAltitude = b'Lald'  
LocalLightingAngle = b'lagl'  
LocalRange = b'LclR'  
Location = b'Lctn'  
Log = b'Log '  
Logarithmic = b'kLog'  
LowerCase = b'LwCs'  
Luminance = b'lmnc'  
Magenta = b'Mgnt'  
MakeVisible = b'MkVs'  
ManipulationFOV = b'Mfov'  
MapBlack = b'MpBl'  
Mapping = b'Mpng'  
MappingShape = b'MpgS'  
Material = b'Mtrl'  
Matrix = b'Mtrx'  
MatteColor = b'MttC'  
Maximum = b'Mxm '  
MaximumStates = b'MxmS'  
MemoryUsagePercent = b'MmrU'  
Merge = b'Mrge'  
Merged = b'Mrgd'  
Message = b'Msge'  
Method = b'Mthd'  
MezzotintType = b'MztT'  
Midpoint = b'Mdpn'



MidtoneLevels = b'MdtL'  
Minimum = b'Mnm '  
MismatchCMYK = b'MsmC'  
MismatchGray = b'MsmG'  
MismatchRGB = b'MsmR'  
Mode = b'Md '  
Monochromatic = b'Mnch'  
MoveTo = b'MvT '  
Name = b'Nm '  
Negative = b'Ngtv'  
New = b'Nw '  
Noise = b'Nose'  
NonImageData = b'NnIm'  
NonLinear = b'NnLn'  
Null = b'null'  
NumLights = b'Nm L'  
Number = b'Nmbr'  
NumberOfCacheLevels = b'NCch'  
NumberOfCacheLevels64 = b'NC64'  
NumberOfChannels = b'NmbO'  
NumberOfChildren = b'NmbC'  
NumberOfDocuments = b'NmbD'  
NumberOfGenerators = b'NmbG'  
NumberOfLayers = b'NmbL'  
NumberOfLevels = b'NmbL'  
NumberOfPaths = b'NmbP'  
NumberOfRipples = b'NmbR'  
NumberOfSiblings = b'NmbS'  
ObjectName = b'ObjN'  
Offset = b'Ofst'  
OldSmallFontType = b'Sftt'  
On = b'On '  
Opacity = b'Opct'  
Optimized = b'Optm'  
Orientation = b'Ornt'  
OriginalHeader = b'OrgH'

OriginalTransmissionReference = b'OrgT'  
OtherCursors = b'OthC'  
OuterGlow = b'OrGl'  
Output = b'Otpt'  
OutputBlackPoint = b'kOBP'  
OutputWhitePoint = b'kOWP'  
OverprintColors = b'OvrC'  
OverrideOpen = b'OvrO'  
OverridePrinter = b'ObrP'  
OverrideSave = b'Ovrd'  
PNGFilter = b'PNGf'  
PNGInterlaceType = b'PGIT'  
PageFormat = b'PMpf'  
PageNumber = b'PgNm'  
PagePosition = b'PgPs'  
PageSetup = b'PgSt'  
PaintCursorKind = b'PnCK'  
PaintType = b'PntT'  
PaintingCursors = b'PntC'  
Palette = b'Plt '  
PaletteFile = b'PltF'  
PaperBrightness = b'PprB'  
ParentIndex = b'PrIn'  
ParentName = b'PrNm'  
Path = b'Path'  
PathContents = b'PthC'  
PathName = b'PthN'  
Pattern = b'Pttn'  
PencilWidth = b'Pncl'  
PerspectiveIndex = b'Prsp'  
Phosphors = b'Phsp'  
PickerID = b'PckI'  
PickerKind = b'Pckr'  
PixelPaintSize = b'PPSz'  
Platform = b'Pltf'  
PluginFolder = b'PlgF'

```
PluginPrefs = b'PlgP'  
Points = b'Pts '  
Position = b'Pstn'  
PostScriptColor = b'PstS'  
Posterization = b'Pstr'  
PredefinedColors = b'PrdC'  
PreferBuiltin = b'PrfB'  
Preferences = b'Prfr'  
PreserveAdditional = b'PrsA'  
PreserveLuminosity = b'PrsL'  
PreserveTransparency = b'PrsT'  
Pressure = b'Prs '  
Preview = b'Prvw'  
PreviewCMYK = b'PrvK'  
PreviewFullSize = b'PrvF'  
PreviewIcon = b'PrvI'  
PreviewMacThumbnail = b'PrvM'  
PreviewWinThumbnail = b'PrvW'  
PreviewsQuery = b'PrvQ'  
PrintSettings = b'PMps'  
ProfileSetup = b'PrfS'  
ProvinceState = b'PrvS'  
Quality = b'Qlty'  
QuickMask = b'QucM'  
RGBSetup = b'RGBS'  
Radius = b'Rds '  
RandomSeed = b'RndS'  
Ratio = b'Rt '  
RecentFiles = b'Rcnf'  
Red = b'Rd '  
RedBlackPoint = b'RdB1'  
RedGamma = b'RdGm'  
RedWhitePoint = b'RdWh'  
RedX = b'RdX '  
RedY = b'RdY '  
RegistrationMarks = b'RgsM'
```

Relative = b'Rltv'  
Relief = b'Rlf '  
RenderFidelity = b'Rfid'  
Resample = b'Rsmp'  
ResizeWindowsOnZoom = b'RWOZ'  
Resolution = b'Rslt'  
ResourceID = b'RsrI'  
Response = b'Rspn'  
RetainHeader = b'RtnH'  
Reverse = b'Rvrs'  
Right = b'Rght'  
RippleMagnitude = b'RplM'  
RippleSize = b'Rpls'  
Rotate = b'Rtt '  
Roundness = b'Rndn'  
RulerOriginH = b'RlrH'  
RulerOriginV = b'RlrV'  
RulerUnits = b'RlrU'  
Saturation = b'Strt'  
SaveAndClose = b'SvAn'  
SaveComposite = b'SvCm'  
SavePaletteLocations = b'PltL'  
SavePaths = b'SvPt'  
SavePyramids = b'SvPy'  
Saving = b'Svng'  
Scale = b'Scl '  
ScaleHorizontal = b'SclH'  
ScaleVertical = b'SclV'  
Scaling = b'Scln'  
Scans = b'Scns'  
ScratchDisks = b'ScrD'  
ScreenFile = b'ScrF'  
ScreenType = b'ScrT'  
Separations = b'Sprt'  
SerialString = b'Srls'  
ShadingIntensity = b'ShdI'

ShadingNoise = b'ShdN'  
ShadingShape = b'ShdS'  
ShadowColor = b'sdwC'  
ShadowIntensity = b'ShdI'  
ShadowLevels = b'ShdL'  
ShadowMode = b'sdwM'  
ShadowOpacity = b'sdwO'  
Shape = b'Shp '  
Sharpness = b'Shrp'  
ShearEd = b'ShrE'  
ShearPoints = b'ShrP'  
ShearSt = b'ShrS'  
ShiftKey = b'ShfK'  
ShiftKeyToolSwitch = b'ShKT'  
ShortNames = b'ShrN'  
ShowEnglishFontNames = b'ShwE'  
ShowMenuColors = b'SwMC'  
ShowToolTips = b'ShwT'  
ShowTransparency = b'ShTr'  
SizeKey = b'Sz '  
Skew = b'Skew'  
SmallFontType = b'Sfts'  
SmartBlurMode = b'SmBM'  
SmartBlurQuality = b'SmBQ'  
Smooth = b'Smoo'  
Smoothness = b'Smth'  
SnapshotInitial = b'SnpI'  
SoftClip = b'SfCl'  
Softness = b'Sftn'  
SolidFill = b'SoFi'  
Source = b'Srce'  
Source2 = b'Src2'  
SourceMode = b'SrcM'  
Spacing = b'Spcn'  
SpecialInstructions = b'SpcI'  
SpherizeMode = b'SphM'

```
Spot = b'Spot'  
SprayRadius = b'SprR'  
SquareSize = b'SqrS'  
SrcBlackMax = b'SrcL'  
SrcBlackMin = b'SrcB'  
SrcWhiteMax = b'Srcm'  
SrcWhiteMin = b'SrcW'  
Start = b'Strt'  
StartArrowhead = b'StrA'  
State = b'Stte'  
Strength = b'srgh'  
StrengthRatio = b'srgR'  
Strength_PLUGIN = b'Strg'  
StrokeDetail = b'StDt'  
StrokeDirection = b'SDir'  
StrokeLength = b'StrL'  
StrokePressure = b'StrP'  
StrokeSize = b'StrS'  
StrokeWidth = b'StrW'  
Style = b'Styl'  
Styles = b'Stys'  
StylusIsColor = b'StLC'  
StylusIsOpacity = b'StLO'  
StylusIsPressure = b'StLP'  
StylusIsSize = b'StLS'  
SubPathList = b'SbpL'  
SupplementalCategories = b'SplC'  
SystemInfo = b'SstI'  
SystemPalette = b'SstP'  
Target = b'null'  
TargetPath = b'Trgp'  
TargetPathIndex = b'TrgP'  
TermLength = b'Lngt'  
Text = b'Txt '  
TextClickPoint = b'TxtC'  
TextData = b'TxtD'
```

TextStyle = b'TxtS'  
TextStyleRange = b'Txtt'  
Texture = b'Txtr'  
TextureCoverage = b'TxtC'  
TextureFile = b'TxtF'  
TextureType = b'TxtT'  
Threshold = b'Thsh'  
TileNumber = b'TlNm'  
TileOffset = b'TlOf'  
TileSize = b'TlSz'  
Title = b'Ttl '  
To = b'T '  
ToBuiltin = b'TBl '  
ToLinked = b'ToLk'  
ToMode = b'TMd '  
ToggleOthers = b'TglO'  
Tolerance = b'Tlrn'  
Top = b'Top '  
TotalLimit = b'Ttll'  
Tracking = b'Trck'  
TransferFunction = b'TrnF'  
TransferSpec = b'TrnS'  
Transparency = b'Trns'  
TransparencyGrid = b'TrnG'  
TransparencyGridColors = b'TrnC'  
TransparencyGridSize = b'TrnG'  
TransparencyPrefs = b'TrnP'  
TransparencyShape = b'TrnS'  
TransparentIndex = b'TrnI'  
TransparentWhites = b'TrnW'  
Twist = b'Twst'  
Type = b'Type'  
UCA = b'UC '  
URL = b'URL '  
UndefinedArea = b'UndA'  
Underline = b'Undl'

```
UnitsPrefs = b'UntP'  
Untitled = b'Untl'  
UpperY = b'UppY'  
Urgency = b'Urgn'  
UseAccurateScreens = b'AcrS'  
UseAdditionalPlugins = b'AdPl'  
UseCacheForHistograms = b'UsCc'  
UseCurves = b'UsCr'  
UseDefault = b'UsDf'  
UseGlobalAngle = b'uglg'  
UseICCPProfile = b'UsIC'  
UseMask = b'UsMs'  
UserMaskEnabled = b'UsrM'  
UserMaskLinked = b'Usrs'  
Using = b'Usng'  
Value = b'Vl '  
Variance = b'Vrnc'  
Vector0 = b'Vct0'  
Vector1 = b'Vct1'  
VectorColor = b'VctC'  
VersionFix = b'VrsF'  
VersionMajor = b'VrsM'  
VersionMinor = b'VrsN'  
Vertical = b'Vrtc'  
VerticalScale = b'VrtS'  
VideoAlpha = b'Vdlp'  
Visible = b'Vsbl'  
WatchSuspension = b'WtcS'  
Watermark = b'watr'  
WaveType = b'Wvtp'  
WavelengthMax = b'WLMx'  
WavelengthMin = b'WLMn'  
WebdavPrefs = b'WbdP'  
WetEdges = b'Wtdg'  
What = b'What'  
WhiteClip = b'WhtC'
```



```

WhiteIntensity = b'WhI'
WhiteIsHigh = b'WhHi'
WhiteLevel = b'WhL'
WhitePoint = b'WhPt'
WholePath = b'WhPt'
Width = b'Wdth'
WindMethod = b'WndM'
With = b'With'
WorkPath = b'WrPt'
WorkPathIndex = b'WrkP'
X = b'X '
Y = b'Y '
Yellow = b'Ylw '
ZigZagType = b'ZZTy'
_3DAntiAlias = b'Alis'
_generate_next_value_ (start, count, last_values)
_member_map_ = {'A': <Key.A: b'A '>, 'Adjustment': <Key.Adjustment: b'Adjs'>, 'Align
_member_names_ = ['_3DAntiAlias', 'A', 'Adjustment', 'Aligned', 'Alignment', 'AllPS',
_member_type_
    alias of builtins.bytes
_value2member_map_ = {b'A ': <Key.A: b'A '>, b'AChn': <Key.AlphaChannelOptions: b'AC
comp = b'comp'

```

### 3.24.6 Type

**class** psd\_tools.terminology.Type

Type definitions extracted from PITerminology.h.

See <https://www.adobe.com/devnet/photoshop/sdk.html>

```

ActionData = b'ActD'
ActionReference = b'#Act'
AlignDistributeSelector = b'ADSt'
Alignment = b'Alg '
Amount = b'Amnt'
AntiAlias = b'Annt'
AreaSelector = b'ArSl'
AssumeOptions = b'AssO'
BevelEmbossStampStyle = b'BESs'
BevelEmbossStyle = b'BESl'

```

```
BitDepth = b'BtDp'  
BlackGeneration = b'BlcG'  
BlendMode = b'BlmM'  
BlurMethod = b'BlrM'  
BlurQuality = b'BlrQ'  
BrushType = b'BrsT'  
BuiltInContour = b'Bltc'  
BuiltinProfile = b'Bltp'  
CMYKSetupEngine = b'CMYE'  
Calculation = b'Clcn'  
Channel = b'Chnl'  
ChannelReference = b'#ChR'  
CheckerboardSize = b'Chck'  
ClassColor = b'#Clr'  
ClassElement = b'#ClE'  
ClassExport = b'#Cle'  
ClassFormat = b'#ClF'  
ClassHueSatHueSatV2 = b'#HsV'  
ClassImport = b'#ClI'  
ClassMode = b'#ClM'  
ClassStringFormat = b'#ClS'  
ClassTextExport = b'#CTE'  
ClassTextImport = b'#ClT'  
Color = b'Clr '  
ColorChannel = b'#ClC'  
ColorPalette = b'ClrP'  
ColorSpace = b'ClrS'  
ColorStopType = b'Clry'  
Colors = b'Clrs'  
Compensation = b'Cmpn'  
ContourEdge = b'CntE'  
Convert = b'Cnvr'  
CorrectionMethod = b'CrcM'  
CursorKind = b'Crsk'  
DCS = b'DCS '  
DeepDepth = b'DpDp'
```

```
Depth = b'Dpth'  
DiffuseMode = b'DfsM'  
Direction = b'Drct'  
DisplacementMap = b'DspM'  
Distribution = b'Dstr'  
Dither = b'Dthr'  
DitherQuality = b'Dthq'  
DocumentReference = b'#DcR'  
EPSPreview = b'EPSP'  
ElementReference = b'#ElR'  
Encoding = b'Encd'  
EraserKind = b'ErsK'  
ExtrudeRandom = b'ExtR'  
ExtrudeType = b'ExtT'  
EyeDropperSample = b'EyDp'  
FPXCompress = b'FxCm'  
Fill = b'Fl '  
FillColor = b'FlCl'  
FillContents = b'FlCn'  
FillMode = b'FlMd'  
ForcedColors = b'FrcC'  
FrameFill = b'FrFl'  
FrameStyle = b'FStl'  
GIFColorFileType = b'GFPT'  
GIFPaletteType = b'GFPL'  
GIFRequiredColorSpaceType = b'GFCS'  
GIFRowOrderType = b'GFIT'  
GlobalClass = b'GlbC'  
GlobalObject = b'GlbO'  
GradientForm = b'GrdF'  
GradientType = b'GrdT'  
GrainType = b'Grnt'  
GrayBehavior = b'GrBh'  
GuideGridColor = b'GdGr'  
GuideGridStyle = b'GdGS'  
HistoryStateSource = b'HstS'
```

```
HorizontalLocation = b'HzL'  
ImageReference = b'#ImR'  
InnerGlowSource = b'IGSr'  
IntegerChannel = b'#inC'  
Intent = b'Inte'  
InterlaceCreateType = b'IntC'  
InterlaceEliminateType = b'IntE'  
Interpolation = b'Intp'  
Kelvin = b'Klvn'  
KelvinCustomWhitePoint = b'#Klv'  
Lens = b'Lns '  
LightDirection = b'LghD'  
LightPosition = b'LghP'  
LightType = b'LghT'  
LocationReference = b'#Lct'  
MaskIndicator = b'MskI'  
MatteColor = b'MttC'  
MatteTechnique = b'BETE'  
MenuItem = b'MnIt'  
Method = b'Mthd'  
MezzotintType = b'MztT'  
Mode = b'Md '  
Notify = b'Ntfy'  
Object = b'Objc'  
ObjectReference = b'obj '  
OnOff = b'OnOf'  
Ordinal = b'Ordn'  
Orientation = b'Ornt'  
PNGFilter = b'PNGf'  
PNGInterlaceType = b'PGIT'  
PagePosition = b'PgPs'  
PathKind = b'PthK'  
PathReference = b'#PtR'  
Phosphors = b'Phsp'  
PhosphorsCustomPhosphors = b'#Phs'  
PickerKind = b'PckK'
```

```
PixelPaintSize = b'PPSz'  
Platform = b'Pltf'  
Preview = b'Prvw'  
PreviewCMYK = b'Prvt'  
ProfileMismatch = b'PrfM'  
PurgeItem = b'PrgI'  
QuadCenterState = b'QCSt'  
Quality = b'Qlty'  
QueryState = b'QurS'  
RGBSetupSource = b'RGBS'  
RawData = b'tdta'  
RippleSize = b'Rpls'  
RulerUnits = b'RlrU'  
ScreenType = b'ScrT'  
Shape = b'Shp '  
SmartBlurMode = b'SmBM'  
SmartBlurQuality = b'SmBQ'  
SourceMode = b'Cndn'  
SpherizeMode = b'SphM'  
State = b'Stte'  
StringChannel = b'#sth'  
StringClassFormat = b'#StC'  
StringCompensation = b'#Stm'  
StringFSS = b'#Stf'  
StringInteger = b'#StI'  
StrokeDirection = b'StrD'  
StrokeLocation = b'StrL'  
TextureType = b'TxtT'  
TransparencyGridColors = b'Trnl'  
TransparencyGridSize = b'TrnG'  
TypeClassModeOrClassMode = b'#TyM'  
UndefinedArea = b'UndA'  
UnitFloat = b'UntF'  
Urgency = b'Urgn'  
UserMaskOptions = b'UsrM'  
ValueList = b'VLLs'
```

```
VerticalLocation = b'VrtL'  
WaveType = b'Wvtp'  
WindMethod = b'WndM'  
YesNo = b'YsN '  
ZigZagType = b'ZZTy'  
_generate_next_value_ (start, count, last_values)  
_member_map_ = {'ActionData': <Type.ActionData: b'ActD'>, 'ActionReference': <Type.  
_member_names_ = ['ActionReference', 'ActionData', 'AlignDistributeSelector', 'Alignme  
_member_type_  
    alias of builtins.bytes  
_value2member_map_ = {b'#Act': <Type.ActionReference: b'#Act'>, b'#CTE': <Type.Class
```

### 3.24.7 Unit

```
class psd_tools.terminology.Unit  
    Unit definitions extracted from PITerminology.h.  
    See https://www.adobe.com/devnet/photoshop/sdk.html  
    Angle = b'#Ang'  
    Density = b'#Rsl'  
    Distance = b'#Rlt'  
    Millimeters = b'#Mlm'  
    Percent = b'#Prc'  
    Pixels = b'#Pxl'  
    Points = b'#Pnt'  
    _None = b'#Nne'  
    _generate_next_value_ (start, count, last_values)  
    _member_map_ = {'Angle': <Unit.Angle: b'#Ang'>, 'Density': <Unit.Density: b'#Rsl'>  
    _member_names_ = ['Angle', 'Density', 'Distance', '_None', 'Percent', 'Pixels', 'Milli  
    _member_type_  
        alias of builtins.bytes  
    _value2member_map_ = {b'#Ang': <Unit.Angle: b'#Ang'>, b'#Mlm': <Unit.Millimeters: }
```

## CHAPTER 4

---

### Indices and tables

---

- `genindex`
- `modindex`
- `search`





**p**

`psd_tools.api.adjustments`, 17  
`psd_tools.api.effects`, 34  
`psd_tools.api.layers`, 41  
`psd_tools.api.mask`, 66  
`psd_tools.api.shape`, 67  
`psd_tools.api.smart_object`, 72  
`psd_tools.constants`, 73  
`psd_tools.psd`, 84  
`psd_tools.psd.base`, 84  
`psd_tools.psd.color_mode_data`, 86  
`psd_tools.psd.descriptor`, 87  
`psd_tools.psd.effects_layer`, 93  
`psd_tools.psd.engine_data`, 92  
`psd_tools.psd.filter_effects`, 96  
`psd_tools.psd.header`, 97  
`psd_tools.psd.image_data`, 98  
`psd_tools.psd.image_resources`, 99  
`psd_tools.psd.layer_and_mask`, 106  
`psd_tools.psd.linked_layer`, 112  
`psd_tools.psd.patterns`, 113  
`psd_tools.psd.tagged_blocks`, 114  
`psd_tools.psd.vector`, 118  
`psd_tools.terminology`, 120



## Symbols

- `_16BitsPerPixel` (*psd\_tools.terminology.Enum attribute*), 144
- `_1BitPerPixel` (*psd\_tools.terminology.Enum attribute*), 144
- `_2BitsPerPixel` (*psd\_tools.terminology.Enum attribute*), 144
- `_32BitsPerPixel` (*psd\_tools.terminology.Enum attribute*), 144
- `_3DAntiAlias` (*psd\_tools.terminology.Key attribute*), 173
- `_3DTransform` (*psd\_tools.terminology.Event attribute*), 150
- `_4BitsPerPixel` (*psd\_tools.terminology.Enum attribute*), 144
- `_5000` (*psd\_tools.terminology.Enum attribute*), 144
- `_5500` (*psd\_tools.terminology.Enum attribute*), 144
- `_6500` (*psd\_tools.terminology.Enum attribute*), 144
- `_72Color` (*psd\_tools.terminology.Enum attribute*), 144
- `_72Gray` (*psd\_tools.terminology.Enum attribute*), 144
- `_7500` (*psd\_tools.terminology.Enum attribute*), 144
- `_8BitsPerPixel` (*psd\_tools.terminology.Enum attribute*), 144
- `_9300` (*psd\_tools.terminology.Enum attribute*), 144
- `_None` (*psd\_tools.terminology.Enum attribute*), 144
- `_None` (*psd\_tools.terminology.Unit attribute*), 178
- `_generate_next_value_()` (*psd\_tools.terminology.Enum method*), 144
- `_generate_next_value_()` (*psd\_tools.terminology.Event method*), 150
- `_generate_next_value_()` (*psd\_tools.terminology.Form method*), 151
- `_generate_next_value_()` (*psd\_tools.terminology.Key method*), 173
- `_generate_next_value_()` (*psd\_tools.terminology.Klass method*), 126
- `_generate_next_value_()` (*psd\_tools.terminology.Type method*), 178
- `_generate_next_value_()` (*psd\_tools.terminology.Unit method*), 178
- `_member_map_` (*psd\_tools.terminology.Enum attribute*), 144
- `_member_map_` (*psd\_tools.terminology.Event attribute*), 150
- `_member_map_` (*psd\_tools.terminology.Form attribute*), 151
- `_member_map_` (*psd\_tools.terminology.Key attribute*), 173
- `_member_map_` (*psd\_tools.terminology.Klass attribute*), 126
- `_member_map_` (*psd\_tools.terminology.Type attribute*), 178
- `_member_map_` (*psd\_tools.terminology.Unit attribute*), 178
- `_member_names_` (*psd\_tools.terminology.Enum attribute*), 144
- `_member_names_` (*psd\_tools.terminology.Event attribute*), 150
- `_member_names_` (*psd\_tools.terminology.Form attribute*), 151
- `_member_names_` (*psd\_tools.terminology.Key attribute*), 173
- `_member_names_` (*psd\_tools.terminology.Klass attribute*), 126
- `_member_names_` (*psd\_tools.terminology.Type attribute*), 178
- `_member_names_` (*psd\_tools.terminology.Unit attribute*), 178
- `_member_type_` (*psd\_tools.terminology.Enum attribute*), 144
- `_member_type_` (*psd\_tools.terminology.Event attribute*), 150
- `_member_type_` (*psd\_tools.terminology.Form attribute*), 151
- `_member_type_` (*psd\_tools.terminology.Key attribute*), 173
- `_member_type_` (*psd\_tools.terminology.Klass attribute*), 126
- `_member_type_` (*psd\_tools.terminology.Type attribute*), 178

tribute), 178  
 \_member\_type\_ (*psd\_tools.terminology.Unit attribute*), 178  
 \_value2member\_map\_ (*psd\_tools.terminology.Enum attribute*), 144  
 \_value2member\_map\_ (*psd\_tools.terminology.Event attribute*), 151  
 \_value2member\_map\_ (*psd\_tools.terminology.Form attribute*), 151  
 \_value2member\_map\_ (*psd\_tools.terminology.Key attribute*), 173  
 \_value2member\_map\_ (*psd\_tools.terminology.Klass attribute*), 126  
 \_value2member\_map\_ (*psd\_tools.terminology.Type attribute*), 178  
 \_value2member\_map\_ (*psd\_tools.terminology.Unit attribute*), 178

## A

A (*psd\_tools.terminology.Enum attribute*), 126  
 A (*psd\_tools.terminology.Key attribute*), 151  
 AboutApp (*psd\_tools.terminology.Enum attribute*), 127  
 AbsColorimetric (*psd\_tools.terminology.Enum attribute*), 127  
 Absolute (*psd\_tools.terminology.Enum attribute*), 127  
 AccentedEdges (*psd\_tools.terminology.Event attribute*), 144  
 Action (*psd\_tools.terminology.Klass attribute*), 121  
 ActionData (*psd\_tools.terminology.Type attribute*), 173  
 ActionReference (*psd\_tools.terminology.Type attribute*), 173  
 ActionSet (*psd\_tools.terminology.Klass attribute*), 121  
 ActualPixels (*psd\_tools.terminology.Enum attribute*), 127  
 Adaptive (*psd\_tools.terminology.Enum attribute*), 127  
 Add (*psd\_tools.terminology.Enum attribute*), 127  
 Add (*psd\_tools.terminology.Event attribute*), 144  
 AddNoise (*psd\_tools.terminology.Event attribute*), 144  
 AddTo (*psd\_tools.terminology.Event attribute*), 144  
 Adjustment (*psd\_tools.terminology.Key attribute*), 151  
 Adjustment (*psd\_tools.terminology.Klass attribute*), 121  
 AdjustmentLayer (*psd\_tools.terminology.Klass attribute*), 121  
 AdjustmentOptions (*psd\_tools.terminology.Enum attribute*), 127  
 AdobeRGB1998 (*psd\_tools.terminology.Enum attribute*), 127  
 ADSBottoms (*psd\_tools.terminology.Enum attribute*), 126

ADSCentersH (*psd\_tools.terminology.Enum attribute*), 126  
 ADSCentersV (*psd\_tools.terminology.Enum attribute*), 126  
 ADSHorizontal (*psd\_tools.terminology.Enum attribute*), 126  
 ADSLefts (*psd\_tools.terminology.Enum attribute*), 126  
 ADSRights (*psd\_tools.terminology.Enum attribute*), 126  
 ADSTops (*psd\_tools.terminology.Enum attribute*), 126  
 ADSVertical (*psd\_tools.terminology.Enum attribute*), 127  
 AirbrushEraser (*psd\_tools.terminology.Enum attribute*), 127  
 AirbrushTool (*psd\_tools.terminology.Klass attribute*), 121  
 Alias (*class in psd\_tools.psd.descriptor*), 87  
 ALIAS (*psd\_tools.constants.LinkedLayerType attribute*), 76  
 Align (*psd\_tools.terminology.Event attribute*), 144  
 AlignDistributeSelector (*psd\_tools.terminology.Type attribute*), 173  
 aligned (*psd\_tools.api.effects.GradientOverlay attribute*), 37  
 aligned (*psd\_tools.api.effects.PatternOverlay attribute*), 38  
 Aligned (*psd\_tools.terminology.Key attribute*), 151  
 Alignment (*psd\_tools.terminology.Key attribute*), 151  
 Alignment (*psd\_tools.terminology.Type attribute*), 173  
 All (*psd\_tools.terminology.Enum attribute*), 127  
 All (*psd\_tools.terminology.Event attribute*), 144  
 AllExcept (*psd\_tools.terminology.Key attribute*), 151  
 AllPS (*psd\_tools.terminology.Key attribute*), 151  
 AllToolOptions (*psd\_tools.terminology.Key attribute*), 151  
 ALPHA (*psd\_tools.constants.Tag attribute*), 81  
 alpha (*psd\_tools.psd.image\_resources.SliceV6 attribute*), 105  
 ALPHA\_IDENTIFIERS (*psd\_tools.constants.Resource attribute*), 77  
 ALPHA\_NAMES\_PASCAL (*psd\_tools.constants.Resource attribute*), 77  
 ALPHA\_NAMES\_UNICODE (*psd\_tools.constants.Resource attribute*), 77  
 AlphaChannelOptions (*psd\_tools.terminology.Key attribute*), 151  
 AlphaChannelOptions (*psd\_tools.terminology.Klass attribute*), 121  
 AlphaChannels (*psd\_tools.terminology.Key attribute*), 151  
 AlphaIdentifiers (*class in psd\_tools.psd.image\_resources*), 101

- AlphaNamesPascal (class *psd\_tools.psd.image\_resources*), 101
- AlphaNamesUnicode (class *psd\_tools.psd.image\_resources*), 101
- alt\_tag (*psd\_tools.psd.image\_resources.SliceV6* attribute), 104
- ALTERNATE\_DUOTONE\_COLORS (*psd\_tools.constants.Resource* attribute), 77
- ALTERNATE\_SPOT\_COLORS (*psd\_tools.constants.Resource* attribute), 77
- altitude (*psd\_tools.api.effects.BevelEmboss* attribute), 39
- AmbientBrightness (*psd\_tools.terminology.Key* attribute), 151
- AmbientColor (*psd\_tools.terminology.Key* attribute), 151
- Amiga (*psd\_tools.terminology.Enum* attribute), 127
- Amount (*psd\_tools.terminology.Key* attribute), 151
- Amount (*psd\_tools.terminology.Type* attribute), 173
- AmountHigh (*psd\_tools.terminology.Enum* attribute), 127
- AmountLow (*psd\_tools.terminology.Enum* attribute), 127
- AmountMedium (*psd\_tools.terminology.Enum* attribute), 127
- AmplitudeMax (*psd\_tools.terminology.Key* attribute), 151
- AmplitudeMin (*psd\_tools.terminology.Key* attribute), 152
- anchor (*psd\_tools.psd.vector.Knot* attribute), 119
- Anchor (*psd\_tools.terminology.Key* attribute), 152
- angle (*psd\_tools.api.effects.BevelEmboss* attribute), 39
- angle (*psd\_tools.api.effects.DropShadow* attribute), 34
- angle (*psd\_tools.api.effects.GradientOverlay* attribute), 38
- angle (*psd\_tools.api.effects.InnerShadow* attribute), 35
- angle (*psd\_tools.api.effects.Satin* attribute), 41
- angle (*psd\_tools.psd.effects\_layer.BevelInfo* attribute), 95
- angle (*psd\_tools.psd.effects\_layer.ShadowInfo* attribute), 94
- angle (*psd\_tools.psd.image\_resources.HalftoneScreen* attribute), 102
- Angle (*psd\_tools.terminology.Enum* attribute), 127
- Angle (*psd\_tools.terminology.Key* attribute), 152
- Angle (*psd\_tools.terminology.Unit* attribute), 178
- Angle1 (*psd\_tools.terminology.Key* attribute), 152
- Angle2 (*psd\_tools.terminology.Key* attribute), 152
- Angle3 (*psd\_tools.terminology.Key* attribute), 152
- Angle4 (*psd\_tools.terminology.Key* attribute), 152
- AngledStrokes (*psd\_tools.terminology.Event* attribute), 144
- ANIMATION\_EFFECTS (*psd\_tools.constants.Tag* attribute), 81
- Annotation (class in *psd\_tools.psd.tagged\_blocks*), 115
- Annotations (class in *psd\_tools.psd.tagged\_blocks*), 115
- ANNOTATIONS (*psd\_tools.constants.Tag* attribute), 81
- anti\_aliased (*psd\_tools.api.effects.BevelEmboss* attribute), 39
- anti\_aliased (*psd\_tools.api.effects.DropShadow* attribute), 34
- anti\_aliased (*psd\_tools.api.effects.InnerGlow* attribute), 36
- anti\_aliased (*psd\_tools.api.effects.InnerShadow* attribute), 35
- anti\_aliased (*psd\_tools.api.effects.OuterGlow* attribute), 35
- anti\_aliased (*psd\_tools.api.effects.Satin* attribute), 41
- AntiAlias (*psd\_tools.terminology.Key* attribute), 152
- AntiAlias (*psd\_tools.terminology.Type* attribute), 173
- AntiAliasCrisp (*psd\_tools.terminology.Enum* attribute), 127
- AntiAliasedPICTAcquire (*psd\_tools.terminology.Klass* attribute), 121
- AntiAliasHigh (*psd\_tools.terminology.Enum* attribute), 127
- AntiAliasLow (*psd\_tools.terminology.Enum* attribute), 127
- AntiAliasMedium (*psd\_tools.terminology.Enum* attribute), 127
- AntiAliasNone (*psd\_tools.terminology.Enum* attribute), 127
- AntiAliasSmooth (*psd\_tools.terminology.Enum* attribute), 127
- AntiAliasStrong (*psd\_tools.terminology.Enum* attribute), 127
- Any (*psd\_tools.terminology.Enum* attribute), 127
- Append (*psd\_tools.terminology.Key* attribute), 152
- AppleRGB (*psd\_tools.terminology.Enum* attribute), 127
- Application (*psd\_tools.terminology.Klass* attribute), 121
- Apply (*psd\_tools.terminology.Key* attribute), 152
- ApplyImage (*psd\_tools.terminology.Enum* attribute), 127
- ApplyImage (*psd\_tools.terminology.Event* attribute), 144
- ApplyStyle (*psd\_tools.terminology.Event* attribute), 145
- Area (*psd\_tools.terminology.Key* attribute), 152
- AreaSelector (*psd\_tools.terminology.Type* attribute), 173
- AroundCenter (*psd\_tools.terminology.Enum* attribute), 127

- Arrange (*psd\_tools.terminology.Enum attribute*), 127
- arrow\_conc (*psd\_tools.api.shape.Line attribute*), 69
- arrow\_end (*psd\_tools.api.shape.Line attribute*), 69
- arrow\_length (*psd\_tools.api.shape.Line attribute*), 69
- arrow\_start (*psd\_tools.api.shape.Line attribute*), 69
- arrow\_width (*psd\_tools.api.shape.Line attribute*), 69
- Arrowhead (*psd\_tools.terminology.Key attribute*), 152
- Arrowhead (*psd\_tools.terminology.Klass attribute*), 121
- Artboard (*class in psd\_tools.api.layers*), 41
- ARTBOARD\_DATA1 (*psd\_tools.constants.Tag attribute*), 81
- ARTBOARD\_DATA2 (*psd\_tools.constants.Tag attribute*), 81
- ARTBOARD\_DATA3 (*psd\_tools.constants.Tag attribute*), 81
- ArtHistoryBrushTool (*psd\_tools.terminology.Klass attribute*), 121
- As (*psd\_tools.terminology.Key attribute*), 152
- ASCII (*psd\_tools.terminology.Enum attribute*), 127
- Ask (*psd\_tools.terminology.Enum attribute*), 127
- AskWhenOpening (*psd\_tools.terminology.Enum attribute*), 127
- Assert (*psd\_tools.terminology.Event attribute*), 145
- Assert (*psd\_tools.terminology.Klass attribute*), 121
- AssetBin (*psd\_tools.terminology.Key attribute*), 152
- associated\_id (*psd\_tools.psd.image\_resources.SliceV6 attribute*), 104
- AssumedCMYK (*psd\_tools.terminology.Key attribute*), 152
- AssumedGray (*psd\_tools.terminology.Key attribute*), 152
- AssumedProfile (*psd\_tools.terminology.Klass attribute*), 121
- AssumedRGB (*psd\_tools.terminology.Key attribute*), 152
- AssumeOptions (*psd\_tools.terminology.Type attribute*), 173
- At (*psd\_tools.terminology.Key attribute*), 152
- Auto (*psd\_tools.terminology.Key attribute*), 152
- AUTO\_SAVE\_FILE\_PATH (*psd\_tools.constants.Resource attribute*), 77
- AUTO\_SAVE\_FORMAT (*psd\_tools.constants.Resource attribute*), 77
- AutoContrast (*psd\_tools.terminology.Key attribute*), 152
- AutoErase (*psd\_tools.terminology.Key attribute*), 152
- AutoKern (*psd\_tools.terminology.Key attribute*), 152
- automatic (*psd\_tools.api.adjustments.BrightnessContrast attribute*), 29
- AutoUpdate (*psd\_tools.terminology.Key attribute*), 152
- Average (*psd\_tools.terminology.Event attribute*), 145
- Axis (*psd\_tools.terminology.Key attribute*), 152
- ## B
- B (*psd\_tools.terminology.Enum attribute*), 127
- B (*psd\_tools.terminology.Key attribute*), 152
- Back (*psd\_tools.terminology.Enum attribute*), 127
- Background (*psd\_tools.terminology.Enum attribute*), 127
- Background (*psd\_tools.terminology.Key attribute*), 152
- background\_color (*psd\_tools.api.mask.Mask attribute*), 66
- BACKGROUND\_COLOR (*psd\_tools.constants.Resource attribute*), 77
- background\_color (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 109
- BackgroundColor (*psd\_tools.terminology.Enum attribute*), 127
- BackgroundColor (*psd\_tools.terminology.Key attribute*), 152
- BackgroundEraserTool (*psd\_tools.terminology.Klass attribute*), 121
- BackgroundLayer (*psd\_tools.terminology.Klass attribute*), 121
- BackgroundLevel (*psd\_tools.terminology.Key attribute*), 152
- BackLight (*psd\_tools.terminology.Event attribute*), 145
- BackLight (*psd\_tools.terminology.Klass attribute*), 121
- Backward (*psd\_tools.terminology.Enum attribute*), 127
- Backward (*psd\_tools.terminology.Key attribute*), 152
- Balance (*psd\_tools.terminology.Key attribute*), 152
- BASE (*psd\_tools.constants.Clipping attribute*), 75
- BaseElement (*class in psd\_tools.psd.base*), 85
- BaselineShift (*psd\_tools.terminology.Key attribute*), 152
- BasRelief (*psd\_tools.terminology.Event attribute*), 145
- Batch (*psd\_tools.terminology.Event attribute*), 145
- BatchFromDroplet (*psd\_tools.terminology.Event attribute*), 145
- bbox (*psd\_tools.api.adjustments.GradientFill attribute*), 25
- bbox (*psd\_tools.api.adjustments.PatternFill attribute*), 21
- bbox (*psd\_tools.api.adjustments.SolidColorFill attribute*), 17
- bbox (*psd\_tools.api.layers.Artboard attribute*), 41
- bbox (*psd\_tools.api.layers.Group attribute*), 46
- bbox (*psd\_tools.api.layers.PixelLayer attribute*), 50
- bbox (*psd\_tools.api.layers.ShapeLayer attribute*), 54



- bbox** (*psd\_tools.api.layers.SmartObjectLayer* attribute), 58  
**bbox** (*psd\_tools.api.layers.TypeLayer* attribute), 62  
**bbox** (*psd\_tools.api.mask.Mask* attribute), 66  
**bbox** (*psd\_tools.api.shape.Ellipse* attribute), 70  
**bbox** (*psd\_tools.api.shape.Line* attribute), 70  
**bbox** (*psd\_tools.api.shape.Rectangle* attribute), 71  
**bbox** (*psd\_tools.api.shape.RoundedRectangle* attribute), 72  
**bbox** (*psd\_tools.api.shape.VectorMask* attribute), 67  
**bbox** (*psd\_tools.psd.image\_resources.SlicesV6* attribute), 104  
**bbox** (*psd\_tools.psd.image\_resources.SliceV6* attribute), 104  
**bbox** (*psd\_tools.PSDImage* attribute), 12  
**BeepWhenDone** (*psd\_tools.terminology.Key* attribute), 152  
**BeginRamp** (*psd\_tools.terminology.Key* attribute), 152  
**BeginSustain** (*psd\_tools.terminology.Key* attribute), 152  
**Behind** (*psd\_tools.terminology.Enum* attribute), 127  
**Best** (*psd\_tools.terminology.Enum* attribute), 128  
**Better** (*psd\_tools.terminology.Enum* attribute), 128  
**BEVEL** (*psd\_tools.constants.EffectOSType* attribute), 76  
**bevel\_style** (*psd\_tools.api.effects.BevelEmboss* attribute), 40  
**bevel\_type** (*psd\_tools.api.effects.BevelEmboss* attribute), 40  
**BevelDirection** (*psd\_tools.terminology.Key* attribute), 152  
**BevelEmboss** (class in *psd\_tools.api.effects*), 39  
**BevelEmboss** (*psd\_tools.terminology.Key* attribute), 152  
**BevelEmboss** (*psd\_tools.terminology.Klass* attribute), 121  
**BevelEmbossStampStyle** (*psd\_tools.terminology.Type* attribute), 173  
**BevelEmbossStyle** (*psd\_tools.terminology.Type* attribute), 173  
**BevelInfo** (class in *psd\_tools.psd.effects\_layer*), 95  
**BevelStyle** (*psd\_tools.terminology.Key* attribute), 153  
**BevelTechnique** (*psd\_tools.terminology.Key* attribute), 153  
**Bicubic** (*psd\_tools.terminology.Enum* attribute), 128  
**BigNudgeH** (*psd\_tools.terminology.Key* attribute), 153  
**BigNudgeV** (*psd\_tools.terminology.Key* attribute), 153  
**Bilinear** (*psd\_tools.terminology.Enum* attribute), 128  
**Binary** (*psd\_tools.terminology.Enum* attribute), 128  
**BitDepth** (*psd\_tools.terminology.Key* attribute), 153  
**BitDepth** (*psd\_tools.terminology.Type* attribute), 174  
**BitDepth1** (*psd\_tools.terminology.Enum* attribute), 128  
**BitDepth16** (*psd\_tools.terminology.Enum* attribute), 128  
**BitDepth24** (*psd\_tools.terminology.Enum* attribute), 128  
**BitDepth32** (*psd\_tools.terminology.Enum* attribute), 128  
**BitDepth4** (*psd\_tools.terminology.Enum* attribute), 128  
**BitDepth8** (*psd\_tools.terminology.Enum* attribute), 128  
**BitDepthA1R5G5B5** (*psd\_tools.terminology.Enum* attribute), 128  
**BitDepthA4R4G4B4** (*psd\_tools.terminology.Enum* attribute), 128  
**BitDepthR5G6B5** (*psd\_tools.terminology.Enum* attribute), 128  
**BitDepthX4R4G4B4** (*psd\_tools.terminology.Enum* attribute), 128  
**BitDepthX8R8G8B8** (*psd\_tools.terminology.Enum* attribute), 128  
**BITMAP** (*psd\_tools.constants.ColorMode* attribute), 75  
**Bitmap** (*psd\_tools.terminology.Enum* attribute), 128  
**BitmapMode** (*psd\_tools.terminology.Klass* attribute), 121  
**bits** (*psd\_tools.psd.image\_resources.ThumbnailResource* attribute), 105  
**Black** (*psd\_tools.terminology.Enum* attribute), 128  
**Black** (*psd\_tools.terminology.Key* attribute), 153  
**BLACK\_AND\_WHITE** (*psd\_tools.constants.Tag* attribute), 81  
**BlackAndWhite** (class in *psd\_tools.api.adjustments*), 31  
**BlackAndWhite** (*psd\_tools.terminology.Enum* attribute), 128  
**BlackBody** (*psd\_tools.terminology.Enum* attribute), 128  
**BlackClip** (*psd\_tools.terminology.Key* attribute), 153  
**BlackGeneration** (*psd\_tools.terminology.Key* attribute), 153  
**BlackGeneration** (*psd\_tools.terminology.Type* attribute), 174  
**BlackGenerationCurve** (*psd\_tools.terminology.Key* attribute), 153  
**BlackIntensity** (*psd\_tools.terminology.Key* attribute), 153  
**BlackLevel** (*psd\_tools.terminology.Key* attribute), 153  
**BlackLimit** (*psd\_tools.terminology.Key* attribute), 153  
**Blacks** (*psd\_tools.terminology.Enum* attribute), 128  
**Blast** (*psd\_tools.terminology.Enum* attribute), 128  
**Bleed** (*psd\_tools.terminology.Key* attribute), 153  
**bleed\_width\_scale** (*psd\_tools.psd.image\_resources.PrintFlagsInfo* attribute), 103

- bleed\_width\_value (psd\_tools.psd.image\_resources.PrintFlagsInfo attribute), 103
- BLEND\_CLIPPING\_ELEMENTS (psd\_tools.constants.Tag attribute), 81
- BLEND\_FILL\_OPACITY (psd\_tools.constants.Tag attribute), 81
- BLEND\_INTERIOR\_ELEMENTS (psd\_tools.constants.Tag attribute), 81
- blend\_mode (psd\_tools.api.adjustments.GradientFill attribute), 25
- blend\_mode (psd\_tools.api.adjustments.PatternFill attribute), 21
- blend\_mode (psd\_tools.api.adjustments.SolidColorFill attribute), 17
- blend\_mode (psd\_tools.api.effects.ColorOverlay attribute), 37
- blend\_mode (psd\_tools.api.effects.DropShadow attribute), 34
- blend\_mode (psd\_tools.api.effects.GradientOverlay attribute), 38
- blend\_mode (psd\_tools.api.effects.InnerGlow attribute), 36
- blend\_mode (psd\_tools.api.effects.InnerShadow attribute), 35
- blend\_mode (psd\_tools.api.effects.OuterGlow attribute), 35
- blend\_mode (psd\_tools.api.effects.PatternOverlay attribute), 38
- blend\_mode (psd\_tools.api.effects.Satin attribute), 41
- blend\_mode (psd\_tools.api.effects.Stroke attribute), 39
- blend\_mode (psd\_tools.api.layers.Artboard attribute), 41
- blend\_mode (psd\_tools.api.layers.Group attribute), 46
- blend\_mode (psd\_tools.api.layers.PixelLayer attribute), 50
- blend\_mode (psd\_tools.api.layers.ShapeLayer attribute), 54
- blend\_mode (psd\_tools.api.layers.SmartObjectLayer attribute), 58
- blend\_mode (psd\_tools.api.layers.TypeLayer attribute), 62
- blend\_mode (psd\_tools.api.shape.Stroke attribute), 68
- blend\_mode (psd\_tools.psd.effects\_layer.InnerGlowInfo attribute), 95
- blend\_mode (psd\_tools.psd.effects\_layer.OuterGlowInfo attribute), 95
- blend\_mode (psd\_tools.psd.effects\_layer.ShadowInfo attribute), 94
- blend\_mode (psd\_tools.psd.effects\_layer.SolidFillInfo attribute), 96
- blend\_mode (psd\_tools.psd.layer\_and\_mask.LayerRecord attribute), 108
- blend\_mode (psd\_tools.psd.tagged\_blocks.SectionDividers attribute), 117
- blending\_ranges (psd\_tools.psd.layer\_and\_mask.LayerRecord attribute), 108
- BlendMode (class in psd\_tools.constants), 73
- BlendMode (psd\_tools.terminology.Type attribute), 174
- BlendRange (psd\_tools.terminology.Key attribute), 153
- BlendRange (psd\_tools.terminology.Klass attribute), 121
- BlockEraser (psd\_tools.terminology.Enum attribute), 128
- Blocks (psd\_tools.terminology.Enum attribute), 128
- blue (psd\_tools.api.adjustments.BlackAndWhite attribute), 31
- blue (psd\_tools.psd.image\_resources.SliceV6 attribute), 105
- Blue (psd\_tools.terminology.Enum attribute), 128
- Blue (psd\_tools.terminology.Key attribute), 153
- BlueBlackPoint (psd\_tools.terminology.Key attribute), 153
- BlueGamma (psd\_tools.terminology.Key attribute), 153
- Blues (psd\_tools.terminology.Enum attribute), 128
- BlueWhitePoint (psd\_tools.terminology.Key attribute), 153
- BlueX (psd\_tools.terminology.Key attribute), 153
- BlueY (psd\_tools.terminology.Key attribute), 153
- blur (psd\_tools.psd.effects\_layer.BevelInfo attribute), 95
- blur (psd\_tools.psd.effects\_layer.InnerGlowInfo attribute), 95
- blur (psd\_tools.psd.effects\_layer.OuterGlowInfo attribute), 94
- blur (psd\_tools.psd.effects\_layer.ShadowInfo attribute), 94
- Blur (psd\_tools.terminology.Event attribute), 145
- Blur (psd\_tools.terminology.Key attribute), 153
- BlurMethod (psd\_tools.terminology.Key attribute), 153
- BlurMethod (psd\_tools.terminology.Type attribute), 174
- BlurMore (psd\_tools.terminology.Event attribute), 145
- BlurQuality (psd\_tools.terminology.Key attribute), 153
- BlurQuality (psd\_tools.terminology.Type attribute), 174
- BlurTool (psd\_tools.terminology.Klass attribute), 121
- BMPFormat (psd\_tools.terminology.Klass attribute), 121
- Book (psd\_tools.terminology.Key attribute), 153
- BookColor (psd\_tools.terminology.Klass attribute), 121
- Bool (class in psd\_tools.psd.descriptor), 87
- Bool (class in psd\_tools.psd.engine\_data), 93
- BooleanElement (class in psd\_tools.psd.base), 86



- Border (*psd\_tools.terminology.Event attribute*), 145
- BORDER\_INFO (*psd\_tools.constants.Resource attribute*), 77
- BorderThickness (*psd\_tools.terminology.Key attribute*), 153
- bottom (*psd\_tools.api.adjustments.GradientFill attribute*), 25
- bottom (*psd\_tools.api.adjustments.PatternFill attribute*), 21
- bottom (*psd\_tools.api.adjustments.SolidColorFill attribute*), 17
- bottom (*psd\_tools.api.layers.Artboard attribute*), 42
- bottom (*psd\_tools.api.layers.PixelLayer attribute*), 50
- bottom (*psd\_tools.api.layers.ShapeLayer attribute*), 54
- bottom (*psd\_tools.api.layers.SmartObjectLayer attribute*), 58
- bottom (*psd\_tools.api.layers.TypeLayer attribute*), 62
- bottom (*psd\_tools.api.mask.Mask attribute*), 66
- bottom (*psd\_tools.psd.layer\_and\_mask.LayerRecord attribute*), 108
- bottom (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 109
- bottom (*psd\_tools.psd.tagged\_blocks.TypeToolObjectSetting attribute*), 118
- bottom (*psd\_tools.psd.vector.ClipboardRecord attribute*), 119
- bottom (*psd\_tools.PSDImage attribute*), 12
- Bottom (*psd\_tools.terminology.Enum attribute*), 128
- Bottom (*psd\_tools.terminology.Key attribute*), 153
- BOUNDING\_SECTION\_DIVIDER (*psd\_tools.constants.SectionDivider attribute*), 81
- brightness (*psd\_tools.api.adjustments.BrightnessContrast attribute*), 29
- Brightness (*psd\_tools.terminology.Event attribute*), 145
- Brightness (*psd\_tools.terminology.Key attribute*), 153
- BRIGHTNESS\_AND\_CONTRAST (*psd\_tools.constants.Tag attribute*), 81
- BrightnessContrast (*class in psd\_tools.api.adjustments*), 29
- BrightnessContrast (*psd\_tools.terminology.Klass attribute*), 121
- Brush (*psd\_tools.terminology.Klass attribute*), 121
- BrushDarkRough (*psd\_tools.terminology.Enum attribute*), 128
- BrushDetail (*psd\_tools.terminology.Key attribute*), 153
- Brushes (*psd\_tools.terminology.Key attribute*), 153
- BrushesAppend (*psd\_tools.terminology.Enum attribute*), 128
- BrushesDefine (*psd\_tools.terminology.Enum attribute*), 128
- BrushesDelete (*psd\_tools.terminology.Enum attribute*), 129
- BrushesLoad (*psd\_tools.terminology.Enum attribute*), 129
- BrushesNew (*psd\_tools.terminology.Enum attribute*), 129
- BrushesOptions (*psd\_tools.terminology.Enum attribute*), 129
- BrushesReset (*psd\_tools.terminology.Enum attribute*), 129
- BrushesSave (*psd\_tools.terminology.Enum attribute*), 129
- BrushLightRough (*psd\_tools.terminology.Enum attribute*), 128
- BrushSimple (*psd\_tools.terminology.Enum attribute*), 128
- BrushSize (*psd\_tools.terminology.Enum attribute*), 128
- BrushSize (*psd\_tools.terminology.Key attribute*), 153
- BrushSparkle (*psd\_tools.terminology.Enum attribute*), 128
- BrushType (*psd\_tools.terminology.Key attribute*), 153
- BrushType (*psd\_tools.terminology.Type attribute*), 174
- BrushWideBlurry (*psd\_tools.terminology.Enum attribute*), 128
- BrushWideSharp (*psd\_tools.terminology.Enum attribute*), 128
- Builtin (*psd\_tools.terminology.Enum attribute*), 129
- BuiltInContour (*psd\_tools.terminology.Type attribute*), 174
- BuiltInProfile (*psd\_tools.terminology.Type attribute*), 174
- BumpAmplitude (*psd\_tools.terminology.Key attribute*), 153
- BumpChannel (*psd\_tools.terminology.Key attribute*), 153
- BurnInH (*psd\_tools.terminology.Enum attribute*), 129
- BurnInM (*psd\_tools.terminology.Enum attribute*), 129
- BurnInS (*psd\_tools.terminology.Enum attribute*), 129
- BurnInTool (*psd\_tools.terminology.Klass attribute*), 121
- ButtonMode (*psd\_tools.terminology.Enum attribute*), 129
- By (*psd\_tools.terminology.Key attribute*), 153
- Byline (*psd\_tools.terminology.Key attribute*), 153
- BylineTitle (*psd\_tools.terminology.Key attribute*), 153
- Byte (*class in psd\_tools.psd.image\_resources*), 101
- ByteElement (*class in psd\_tools.psd.base*), 86
- ByteOrder (*psd\_tools.terminology.Key attribute*), 154
- Bytes (*class in psd\_tools.psd.tagged\_blocks*), 115

## C

- CachePrefs (*psd\_tools.terminology.Key attribute*),

- 154
- CachePrefs (*psd\_tools.terminology.Klass attribute*), 121
- Calculation (*psd\_tools.terminology.Key attribute*), 154
- Calculation (*psd\_tools.terminology.Klass attribute*), 121
- Calculation (*psd\_tools.terminology.Type attribute*), 174
- Calculations (*psd\_tools.terminology.Enum attribute*), 129
- CalibrationBars (*psd\_tools.terminology.Key attribute*), 154
- CanvasSize (*psd\_tools.terminology.Event attribute*), 145
- Caption (*psd\_tools.terminology.Key attribute*), 154
- CAPTION\_DIGEST (*psd\_tools.constants.Resource attribute*), 77
- CAPTION\_PASCAL (*psd\_tools.constants.Resource attribute*), 77
- CaptionWriter (*psd\_tools.terminology.Key attribute*), 154
- Cascade (*psd\_tools.terminology.Enum attribute*), 129
- Category (*psd\_tools.terminology.Key attribute*), 154
- cell\_is\_html (*psd\_tools.psd.image\_resources.SliceV6 attribute*), 104
- cell\_text (*psd\_tools.psd.image\_resources.SliceV6 attribute*), 104
- CellSize (*psd\_tools.terminology.Key attribute*), 154
- Center (*psd\_tools.terminology.Enum attribute*), 129
- Center (*psd\_tools.terminology.Key attribute*), 154
- center\_crop (*psd\_tools.psd.image\_resources.PrintFlagsInfo attribute*), 103
- CenterCropMarks (*psd\_tools.terminology.Key attribute*), 154
- CENTERED (*psd\_tools.constants.PrintScaleStyle attribute*), 77
- CenteredFrame (*psd\_tools.terminology.Enum attribute*), 129
- CenterGlow (*psd\_tools.terminology.Enum attribute*), 129
- ChalkArea (*psd\_tools.terminology.Key attribute*), 154
- ChalkCharcoal (*psd\_tools.terminology.Event attribute*), 145
- Channel (*psd\_tools.terminology.Key attribute*), 154
- Channel (*psd\_tools.terminology.Klass attribute*), 121
- Channel (*psd\_tools.terminology.Type attribute*), 174
- CHANNEL\_0 (*psd\_tools.constants.ChannelID attribute*), 74
- CHANNEL\_1 (*psd\_tools.constants.ChannelID attribute*), 74
- CHANNEL\_2 (*psd\_tools.constants.ChannelID attribute*), 74
- CHANNEL\_3 (*psd\_tools.constants.ChannelID attribute*), 74
- CHANNEL\_4 (*psd\_tools.constants.ChannelID attribute*), 74
- CHANNEL\_5 (*psd\_tools.constants.ChannelID attribute*), 74
- CHANNEL\_6 (*psd\_tools.constants.ChannelID attribute*), 74
- CHANNEL\_7 (*psd\_tools.constants.ChannelID attribute*), 74
- CHANNEL\_8 (*psd\_tools.constants.ChannelID attribute*), 74
- CHANNEL\_9 (*psd\_tools.constants.ChannelID attribute*), 74
- CHANNEL\_BLENDING\_RESTRICTIONS\_SETTING (*psd\_tools.constants.Tag attribute*), 81
- channel\_image\_data (*psd\_tools.psd.layer\_and\_mask.LayerInfo attribute*), 107
- channel\_info (*psd\_tools.psd.layer\_and\_mask.LayerRecord attribute*), 108
- CHANNEL\_MIXER (*psd\_tools.constants.Tag attribute*), 82
- channel\_ranges (*psd\_tools.psd.layer\_and\_mask.LayerBlendingRanges attribute*), 109
- channel\_sizes (*psd\_tools.psd.layer\_and\_mask.LayerRecord attribute*), 108
- ChannelBlendingRestrictionsSetting (*class in psd\_tools.psd.tagged\_blocks*), 116
- ChannelData (*class in psd\_tools.psd.layer\_and\_mask*), 111
- ChannelDataList (*class in psd\_tools.psd.layer\_and\_mask*), 111
- ChannelID (*class in psd\_tools.constants*), 74
- ChannelImageData (*class in psd\_tools.psd.layer\_and\_mask*), 111
- ChannelInfo (*class in psd\_tools.psd.layer\_and\_mask*), 111
- ChannelMatrix (*psd\_tools.terminology.Key attribute*), 154
- ChannelMatrix (*psd\_tools.terminology.Klass attribute*), 121
- ChannelMixer (*class in psd\_tools.api.adjustments*), 32
- ChannelMixer (*psd\_tools.terminology.Event attribute*), 145
- ChannelMixer (*psd\_tools.terminology.Klass attribute*), 121
- ChannelName (*psd\_tools.terminology.Key attribute*), 154
- ChannelOptions (*psd\_tools.terminology.Enum attribute*), 129
- ChannelReference (*psd\_tools.terminology.Type attribute*), 174
- channels (*psd\_tools.constants.ColorMode attribute*),

- 75
- channels (*psd\_tools.psd.filter\_effects.FilterEffect attribute*), 97
- channels (*psd\_tools.psd.header.FileHeader attribute*), 98
- channels (*psd\_tools.psd.patterns.VirtualMemoryArrayList attribute*), 114
- channels (*psd\_tools.PSDImage attribute*), 12
- Channels (*psd\_tools.terminology.Key attribute*), 154
- ChannelsInterleaved (*psd\_tools.terminology.Key attribute*), 154
- ChannelsPaletteOptions (*psd\_tools.terminology.Enum attribute*), 129
- Charcoal (*psd\_tools.terminology.Event attribute*), 145
- CharcoalAmount (*psd\_tools.terminology.Key attribute*), 154
- CharcoalArea (*psd\_tools.terminology.Key attribute*), 154
- CheckerboardLarge (*psd\_tools.terminology.Enum attribute*), 129
- CheckerboardMedium (*psd\_tools.terminology.Enum attribute*), 129
- CheckerboardNone (*psd\_tools.terminology.Enum attribute*), 129
- CheckerboardSize (*psd\_tools.terminology.Type attribute*), 174
- CheckerboardSmall (*psd\_tools.terminology.Enum attribute*), 129
- child\_id (*psd\_tools.psd.linked\_layer.LinkedLayer attribute*), 113
- choke (*psd\_tools.api.effects.DropShadow attribute*), 34
- choke (*psd\_tools.api.effects.InnerGlow attribute*), 36
- choke (*psd\_tools.api.effects.InnerShadow attribute*), 35
- choke (*psd\_tools.api.effects.OuterGlow attribute*), 36
- ChokeMatte (*psd\_tools.terminology.Key attribute*), 154
- Chrome (*psd\_tools.terminology.Event attribute*), 145
- ChromeFX (*psd\_tools.terminology.Key attribute*), 154
- ChromeFX (*psd\_tools.terminology.Klass attribute*), 121
- CIERGB (*psd\_tools.terminology.Enum attribute*), 129
- CineonFormat (*psd\_tools.terminology.Klass attribute*), 122
- City (*psd\_tools.terminology.Key attribute*), 154
- Class (*class in psd\_tools.psd.descriptor*), 87
- Class (*psd\_tools.terminology.Form attribute*), 151
- Class1 (*class in psd\_tools.psd.descriptor*), 87
- Class2 (*class in psd\_tools.psd.descriptor*), 88
- Class3 (*class in psd\_tools.psd.descriptor*), 88
- ClassColor (*psd\_tools.terminology.Type attribute*), 174
- ClassElement (*psd\_tools.terminology.Type attribute*), 174
- ClassExport (*psd\_tools.terminology.Type attribute*), 174
- ClassFormat (*psd\_tools.terminology.Type attribute*), 174
- ClassHueSatHueSatV2 (*psd\_tools.terminology.Type attribute*), 174
- classID (*psd\_tools.psd.descriptor.Class attribute*), 87
- classID (*psd\_tools.psd.descriptor.Descriptor attribute*), 88
- classID (*psd\_tools.psd.descriptor.EnumeratedReference attribute*), 89
- classID (*psd\_tools.psd.descriptor.Name attribute*), 90
- classID (*psd\_tools.psd.descriptor.ObjectArray attribute*), 90
- classID (*psd\_tools.psd.descriptor.Offset attribute*), 91
- classID (*psd\_tools.psd.descriptor.Property attribute*), 90
- ClassImport (*psd\_tools.terminology.Type attribute*), 174
- ClassMode (*psd\_tools.terminology.Type attribute*), 174
- ClassStringFormat (*psd\_tools.terminology.Type attribute*), 174
- ClassTextExport (*psd\_tools.terminology.Type attribute*), 174
- ClassTextImport (*psd\_tools.terminology.Type attribute*), 174
- Clear (*psd\_tools.terminology.Enum attribute*), 129
- Clear (*psd\_tools.terminology.Event attribute*), 145
- ClearAmount (*psd\_tools.terminology.Key attribute*), 154
- ClearGuides (*psd\_tools.terminology.Enum attribute*), 129
- clip\_layers (*psd\_tools.api.adjustments.GradientFill attribute*), 25
- clip\_layers (*psd\_tools.api.adjustments.PatternFill attribute*), 21
- clip\_layers (*psd\_tools.api.adjustments.SolidColorFill attribute*), 17
- clip\_layers (*psd\_tools.api.layers.Artboard attribute*), 42
- clip\_layers (*psd\_tools.api.layers.Group attribute*), 46
- clip\_layers (*psd\_tools.api.layers.PixelLayer attribute*), 50
- clip\_layers (*psd\_tools.api.layers.ShapeLayer attribute*), 54
- clip\_layers (*psd\_tools.api.layers.SmartObjectLayer attribute*), 58
- clip\_layers (*psd\_tools.api.layers.TypeLayer attribute*), 62
- CLIPBOARD (*psd\_tools.constants.PathResourceID attribute*), 76
- Clipboard (*psd\_tools.terminology.Enum attribute*), 129
- clipboard\_record (*psd\_tools.api.shape.VectorMask*

- attribute), 67
- ClipboardRecord (class in *psd\_tools.psd.vector*), 119
- Clipping (class in *psd\_tools.constants*), 75
- clipping (*psd\_tools.psd.layer\_and\_mask.LayerRecord* attribute), 108
- CLIPPING\_PATH\_NAME (*psd\_tools.constants.Resource* attribute), 78
- ClippingInfo (*psd\_tools.terminology.Klass* attribute), 122
- ClippingPath (*psd\_tools.terminology.Enum* attribute), 129
- ClippingPath (*psd\_tools.terminology.Key* attribute), 154
- ClippingPath (*psd\_tools.terminology.Klass* attribute), 122
- ClippingPathEPS (*psd\_tools.terminology.Key* attribute), 154
- ClippingPathFlatness (*psd\_tools.terminology.Key* attribute), 154
- ClippingPathIndex (*psd\_tools.terminology.Key* attribute), 154
- ClippingPathInfo (*psd\_tools.terminology.Key* attribute), 154
- CloneSource (*psd\_tools.terminology.Key* attribute), 154
- CloneStampTool (*psd\_tools.terminology.Klass* attribute), 122
- Close (*psd\_tools.terminology.Event* attribute), 145
- CloseAll (*psd\_tools.terminology.Enum* attribute), 129
- CLOSED\_FOLDER (*psd\_tools.constants.SectionDivider* attribute), 81
- CLOSED\_KNOT\_LINKED (*psd\_tools.constants.PathResourceID* attribute), 76
- CLOSED\_KNOT\_UNLINKED (*psd\_tools.constants.PathResourceID* attribute), 76
- CLOSED\_LENGTH (*psd\_tools.constants.PathResourceID* attribute), 76
- ClosedSubpath (*psd\_tools.terminology.Key* attribute), 154
- Clouds (*psd\_tools.terminology.Event* attribute), 145
- CMYK (*psd\_tools.constants.ColorMode* attribute), 75
- CMYK (*psd\_tools.constants.ColorSpaceID* attribute), 75
- CMYK (*psd\_tools.terminology.Enum* attribute), 129
- CMYK64 (*psd\_tools.terminology.Enum* attribute), 129
- CMYKColor (*psd\_tools.terminology.Enum* attribute), 129
- CMYKColor (*psd\_tools.terminology.Klass* attribute), 121
- CMYKColorMode (*psd\_tools.terminology.Klass* attribute), 121
- CMYKSetup (*psd\_tools.terminology.Key* attribute), 154
- CMYKSetup (*psd\_tools.terminology.Klass* attribute), 121
- CMYKSetupEngine (*psd\_tools.terminology.Type* attribute), 174
- CoarseDots (*psd\_tools.terminology.Enum* attribute), 129
- color (*psd\_tools.api.effects.ColorOverlay* attribute), 37
- color (*psd\_tools.api.effects.DropShadow* attribute), 34
- color (*psd\_tools.api.effects.InnerGlow* attribute), 36
- color (*psd\_tools.api.effects.InnerShadow* attribute), 35
- color (*psd\_tools.api.effects.OuterGlow* attribute), 36
- color (*psd\_tools.api.effects.Satin* attribute), 41
- color (*psd\_tools.api.effects.Stroke* attribute), 39
- COLOR (*psd\_tools.constants.BlendMode* attribute), 73
- color (*psd\_tools.psd.effects\_layer.InnerGlowInfo* attribute), 95
- color (*psd\_tools.psd.effects\_layer.OuterGlowInfo* attribute), 95
- color (*psd\_tools.psd.effects\_layer.ShadowInfo* attribute), 94
- color (*psd\_tools.psd.effects\_layer.SolidFillInfo* attribute), 96
- color (*psd\_tools.psd.tagged\_blocks.FilterMask* attribute), 116
- color (*psd\_tools.psd.tagged\_blocks.UserMask* attribute), 118
- Color (*psd\_tools.terminology.Enum* attribute), 129
- Color (*psd\_tools.terminology.Key* attribute), 154
- Color (*psd\_tools.terminology.Klass* attribute), 122
- Color (*psd\_tools.terminology.Type* attribute), 174
- COLOR\_BALANCE (*psd\_tools.constants.Tag* attribute), 82
- COLOR\_BURN (*psd\_tools.constants.BlendMode* attribute), 73
- color\_components (*psd\_tools.api.adjustments.PhotoFilter* attribute), 32
- COLOR\_DODGE (*psd\_tools.constants.BlendMode* attribute), 73
- COLOR\_HALFTONING\_INFO (*psd\_tools.constants.Resource* attribute), 78
- COLOR\_LOOKUP (*psd\_tools.constants.Tag* attribute), 82
- color\_mode (*psd\_tools.psd.header.FileHeader* attribute), 98
- color\_mode (*psd\_tools.PSDImage* attribute), 12
- color\_mode\_data (*psd\_tools.psd.PSD* attribute), 84
- color\_model (*psd\_tools.api.adjustments.GradientMap* attribute), 33
- COLOR\_PROTECTED (*psd\_tools.constants.GlobalLayerMaskKind* attribute), 76
- COLOR\_SAMPLERS\_RESOURCE (*psd\_tools.constants.Resource* attribute), 78



COLOR\_SAMPLERS\_RESOURCE\_OBSOLETE  
(*psd\_tools.constants.Resource* attribute), 78

COLOR\_SELECTED (*psd\_tools.constants.GlobalLayerMaskKind* attribute), 76

color\_space (*psd\_tools.api.adjustments.PhotoFilter* attribute), 32

color\_stops (*psd\_tools.api.adjustments.GradientMap* attribute), 33

color\_table (*psd\_tools.psd.patterns.Pattern* attribute), 113

COLOR\_TRANSFER\_FUNCTION  
(*psd\_tools.constants.Resource* attribute), 78

ColorBalance (class in *psd\_tools.api.adjustments*), 31

ColorBalance (*psd\_tools.terminology.Event* attribute), 145

ColorBalance (*psd\_tools.terminology.Klass* attribute), 122

ColorBurn (*psd\_tools.terminology.Enum* attribute), 129

ColorCast (*psd\_tools.terminology.Event* attribute), 145

ColorCast (*psd\_tools.terminology.Klass* attribute), 122

ColorChannel (*psd\_tools.terminology.Type* attribute), 174

ColorChannels (*psd\_tools.terminology.Key* attribute), 154

ColorCorrection (*psd\_tools.terminology.Key* attribute), 154

ColorCorrection (*psd\_tools.terminology.Klass* attribute), 122

ColorDodge (*psd\_tools.terminology.Enum* attribute), 129

ColoredPencil (*psd\_tools.terminology.Event* attribute), 145

ColorHalftone (*psd\_tools.terminology.Event* attribute), 145

Colorimetric (*psd\_tools.terminology.Enum* attribute), 130

ColorIndicates (*psd\_tools.terminology.Key* attribute), 154

colorization (*psd\_tools.api.adjustments.HueSaturation* attribute), 31

Colorize (*psd\_tools.terminology.Key* attribute), 155

ColorLookup (class in *psd\_tools.api.adjustments*), 32

ColorManagement (*psd\_tools.terminology.Key* attribute), 154

ColorMatch (*psd\_tools.terminology.Enum* attribute), 129

ColorMode (class in *psd\_tools.constants*), 75

ColorModeData (class in *psd\_tools.psd.color\_mode\_data*), 86

ColorNoise (*psd\_tools.terminology.Enum* attribute), 130

ColorOverlay (class in *psd\_tools.api.effects*), 37

ColorPalette (*psd\_tools.terminology.Type* attribute), 174

ColorPickerPrefs (*psd\_tools.terminology.Key* attribute), 154

ColorPickerPrefs (*psd\_tools.terminology.Klass* attribute), 122

ColorRange (*psd\_tools.terminology.Event* attribute), 145

Colors (*psd\_tools.terminology.Key* attribute), 155

Colors (*psd\_tools.terminology.Type* attribute), 174

ColorSampler (*psd\_tools.terminology.Klass* attribute), 122

ColorsList (*psd\_tools.terminology.Key* attribute), 155

ColorSpace (*psd\_tools.terminology.Key* attribute), 155

ColorSpace (*psd\_tools.terminology.Type* attribute), 174

ColorSpaceID (class in *psd\_tools.constants*), 75

ColorStop (*psd\_tools.terminology.Klass* attribute), 122

ColorStopType (*psd\_tools.terminology.Type* attribute), 174

ColorTable (*psd\_tools.terminology.Key* attribute), 155

ColumnWidth (*psd\_tools.terminology.Key* attribute), 155

Command (*psd\_tools.terminology.Klass* attribute), 122

CommandKey (*psd\_tools.terminology.Key* attribute), 155

COMMON\_STATE (*psd\_tools.constants.EffectOSType* attribute), 76

CommonStateInfo (class in *psd\_tools.psd.effects\_layer*), 94

comp (*psd\_tools.terminology.Key* attribute), 173

Compensation (*psd\_tools.terminology.Key* attribute), 155

Compensation (*psd\_tools.terminology.Type* attribute), 174

compose () (in module *psd\_tools*), 16

compose () (*psd\_tools.api.adjustments.GradientFill* method), 25

compose () (*psd\_tools.api.adjustments.PatternFill* method), 22

compose () (*psd\_tools.api.adjustments.SolidColorFill* method), 18

compose () (*psd\_tools.api.layers.Artboard* method), 42

compose () (*psd\_tools.api.layers.Group* method), 46

compose () (*psd\_tools.api.layers.PixelLayer* method), 50

- `compose()` (*psd\_tools.api.layers.ShapeLayer method*), 54
- `compose()` (*psd\_tools.api.layers.SmartObjectLayer method*), 58
- `compose()` (*psd\_tools.api.layers.TypeLayer method*), 62
- `compose()` (*psd\_tools.PSDImage method*), 12
- `Composite` (*psd\_tools.terminology.Enum attribute*), 130
- `composite()` (*psd\_tools.api.adjustments.GradientFill method*), 26
- `composite()` (*psd\_tools.api.adjustments.PatternFill method*), 22
- `composite()` (*psd\_tools.api.adjustments.SolidColorFill method*), 18
- `composite()` (*psd\_tools.api.layers.Artboard method*), 42
- `composite()` (*psd\_tools.api.layers.Group method*), 46
- `composite()` (*psd\_tools.api.layers.PixelLayer method*), 50
- `composite()` (*psd\_tools.api.layers.ShapeLayer method*), 54
- `composite()` (*psd\_tools.api.layers.SmartObjectLayer method*), 58
- `composite()` (*psd\_tools.api.layers.TypeLayer method*), 63
- `composite()` (*psd\_tools.PSDImage method*), 12
- `composite_ranges` (*psd\_tools.psd.layer\_and\_mask.LayerAttribute attribute*), 109
- `COMPOSITOR_INFO` (*psd\_tools.constants.Tag attribute*), 82
- `Compression` (*class in psd\_tools.constants*), 75
- `compression` (*psd\_tools.psd.filter\_effects.FilterEffectChannel attribute*), 97
- `compression` (*psd\_tools.psd.filter\_effects.FilterEffectExtra attribute*), 97
- `compression` (*psd\_tools.psd.image\_data.ImageData attribute*), 98
- `compression` (*psd\_tools.psd.layer\_and\_mask.ChannelData attribute*), 111
- `compression` (*psd\_tools.psd.patterns.VirtualMemoryArray attribute*), 114
- `Compression` (*psd\_tools.terminology.Key attribute*), 155
- `Concavity` (*psd\_tools.terminology.Key attribute*), 155
- `Condition` (*psd\_tools.terminology.Key attribute*), 155
- `Constant` (*psd\_tools.terminology.Key attribute*), 155
- `Constrain` (*psd\_tools.terminology.Key attribute*), 155
- `ConstrainProportions` (*psd\_tools.terminology.Key attribute*), 155
- `ConstructionFOV` (*psd\_tools.terminology.Key attribute*), 155
- `ConteCrayon` (*psd\_tools.terminology.Event attribute*), 145
- `content` (*psd\_tools.api.shape.Stroke attribute*), 68
- `CONTENT_GENERATOR_EXTRA_DATA` (*psd\_tools.constants.Tag attribute*), 82
- `Contiguous` (*psd\_tools.terminology.Key attribute*), 155
- `Continue` (*psd\_tools.terminology.Key attribute*), 155
- `Continuity` (*psd\_tools.terminology.Key attribute*), 155
- `contour` (*psd\_tools.api.effects.BevelEmboss attribute*), 40
- `contour` (*psd\_tools.api.effects.DropShadow attribute*), 34
- `contour` (*psd\_tools.api.effects.InnerGlow attribute*), 36
- `contour` (*psd\_tools.api.effects.InnerShadow attribute*), 35
- `contour` (*psd\_tools.api.effects.OuterGlow attribute*), 36
- `contour` (*psd\_tools.api.effects.Satin attribute*), 41
- `Contour` (*psd\_tools.terminology.Klass attribute*), 122
- `ContourCustom` (*psd\_tools.terminology.Enum attribute*), 130
- `ContourDouble` (*psd\_tools.terminology.Enum attribute*), 130
- `ContourEdge` (*psd\_tools.terminology.Type attribute*), 174
- `ContourGaussian` (*psd\_tools.terminology.Enum attribute*), 130
- `ContouringRanges` (*psd\_tools.terminology.Enum attribute*), 130
- `ContourSingle` (*psd\_tools.terminology.Enum attribute*), 130
- `ContourTriple` (*psd\_tools.terminology.Enum attribute*), 130
- `ContourType` (*psd\_tools.terminology.Key attribute*), 155
- `Contract` (*psd\_tools.terminology.Event attribute*), 145
- `contrast` (*psd\_tools.api.adjustments.BrightnessContrast attribute*), 29
- `contrast` (*psd\_tools.terminology.Key attribute*), 155
- `Convert` (*psd\_tools.terminology.Key attribute*), 155
- `convert` (*psd\_tools.terminology.Type attribute*), 174
- `ConvertMode` (*psd\_tools.terminology.Event attribute*), 145
- `ConvertToCMYK` (*psd\_tools.terminology.Enum attribute*), 130
- `ConvertToGray` (*psd\_tools.terminology.Enum attribute*), 130
- `ConvertToLab` (*psd\_tools.terminology.Enum attribute*), 130
- `ConvertToRGB` (*psd\_tools.terminology.Enum attribute*), 130
- `Copy` (*psd\_tools.terminology.Event attribute*), 145
- `Copy` (*psd\_tools.terminology.Key attribute*), 155
- `CopyEffects` (*psd\_tools.terminology.Event attribute*),

- 145
- CopyMerged (*psd\_tools.terminology.Event attribute*), 145
- Copyright (*psd\_tools.terminology.Key attribute*), 155
- COPYRIGHT\_FLAG (*psd\_tools.constants.Resource attribute*), 78
- CopyrightNotice (*psd\_tools.terminology.Key attribute*), 155
- CopyToLayer (*psd\_tools.terminology.Event attribute*), 145
- CornerCropMarks (*psd\_tools.terminology.Key attribute*), 155
- CorrectionMethod (*psd\_tools.terminology.Type attribute*), 174
- Count (*psd\_tools.terminology.Key attribute*), 155
- COUNT\_INFO (*psd\_tools.constants.Resource attribute*), 78
- CountryName (*psd\_tools.terminology.Key attribute*), 155
- CrackBrightness (*psd\_tools.terminology.Key attribute*), 155
- CrackDepth (*psd\_tools.terminology.Key attribute*), 155
- CrackSpacing (*psd\_tools.terminology.Key attribute*), 155
- Craquelure (*psd\_tools.terminology.Event attribute*), 145
- CreateDroplet (*psd\_tools.terminology.Event attribute*), 145
- CreateDuplicate (*psd\_tools.terminology.Enum attribute*), 130
- CreateInterpolation (*psd\_tools.terminology.Enum attribute*), 130
- CreateLayersFromLayerFX (*psd\_tools.terminology.Key attribute*), 155
- creator (*psd\_tools.psd.linked\_layer.LinkedList attribute*), 113
- Credit (*psd\_tools.terminology.Key attribute*), 155
- Crop (*psd\_tools.terminology.Event attribute*), 145
- Cross (*psd\_tools.terminology.Enum attribute*), 130
- Crosshatch (*psd\_tools.terminology.Event attribute*), 145
- Crossover (*psd\_tools.terminology.Key attribute*), 155
- Crystallize (*psd\_tools.terminology.Event attribute*), 145
- Current (*psd\_tools.terminology.Key attribute*), 155
- CurrentHistoryState (*psd\_tools.terminology.Key attribute*), 155
- CurrentLayer (*psd\_tools.terminology.Enum attribute*), 130
- CurrentLight (*psd\_tools.terminology.Key attribute*), 155
- CurrentToolOptions (*psd\_tools.terminology.Key attribute*), 156
- CursorKind (*psd\_tools.terminology.Type attribute*), 174
- Curve (*psd\_tools.terminology.Key attribute*), 156
- CurveFile (*psd\_tools.terminology.Key attribute*), 156
- CurvePoint (*psd\_tools.terminology.Klass attribute*), 122
- Curves (*class in psd\_tools.api.adjustments*), 30
- CURVES (*psd\_tools.constants.Tag attribute*), 82
- Curves (*psd\_tools.terminology.Event attribute*), 145
- Curves (*psd\_tools.terminology.Klass attribute*), 122
- CurvesAdjustment (*psd\_tools.terminology.Klass attribute*), 122
- Custom (*psd\_tools.terminology.Enum attribute*), 130
- Custom (*psd\_tools.terminology.Event attribute*), 146
- Custom (*psd\_tools.terminology.Key attribute*), 156
- CustomForced (*psd\_tools.terminology.Key attribute*), 156
- CustomMatte (*psd\_tools.terminology.Key attribute*), 156
- CustomPalette (*psd\_tools.terminology.Key attribute*), 156
- CustomPalette (*psd\_tools.terminology.Klass attribute*), 122
- CustomPattern (*psd\_tools.terminology.Enum attribute*), 130
- CustomPhosphors (*psd\_tools.terminology.Klass attribute*), 122
- CustomStops (*psd\_tools.terminology.Enum attribute*), 130
- CustomWhitePoint (*psd\_tools.terminology.Klass attribute*), 122
- Cut (*psd\_tools.terminology.Event attribute*), 146
- Cutout (*psd\_tools.terminology.Event attribute*), 146
- CutToLayer (*psd\_tools.terminology.Event attribute*), 146
- cyan (*psd\_tools.api.adjustments.BlackAndWhite attribute*), 31
- Cyan (*psd\_tools.terminology.Enum attribute*), 130
- Cyan (*psd\_tools.terminology.Key attribute*), 156
- Cyans (*psd\_tools.terminology.Enum attribute*), 130
- ## D
- Dark (*psd\_tools.terminology.Enum attribute*), 130
- DARKEN (*psd\_tools.constants.BlendMode attribute*), 73
- Darken (*psd\_tools.terminology.Enum attribute*), 130
- DarkenOnly (*psd\_tools.terminology.Enum attribute*), 130
- DARKER\_COLOR (*psd\_tools.constants.BlendMode attribute*), 73
- DarkIntensity (*psd\_tools.terminology.Key attribute*), 156
- Darkness (*psd\_tools.terminology.Key attribute*), 156

- DarkStrokes (*psd\_tools.terminology.Event* attribute), 146
- DashedLines (*psd\_tools.terminology.Enum* attribute), 130
- data (*psd\_tools.api.adjustments.ChannelMixer* attribute), 32
- data (*psd\_tools.api.adjustments.Curves* attribute), 30
- data (*psd\_tools.api.adjustments.GradientFill* attribute), 26
- data (*psd\_tools.api.adjustments.HueSaturation* attribute), 31
- data (*psd\_tools.api.adjustments.Levels* attribute), 30
- data (*psd\_tools.api.adjustments.PatternFill* attribute), 22
- data (*psd\_tools.api.adjustments.SelectiveColor* attribute), 33
- data (*psd\_tools.api.adjustments.SolidColorFill* attribute), 18
- data (*psd\_tools.api.smart\_object.SmartObject* attribute), 72
- DATA (*psd\_tools.constants.LinkedLayerType* attribute), 76
- data (*psd\_tools.psd.filter\_effects.FilterEffectChannel* attribute), 97
- data (*psd\_tools.psd.filter\_effects.FilterEffectExtra* attribute), 97
- data (*psd\_tools.psd.image\_data.ImageData* attribute), 98
- data (*psd\_tools.psd.image\_resources.ImageResource* attribute), 101
- data (*psd\_tools.psd.image\_resources.Slices* attribute), 104
- data (*psd\_tools.psd.image\_resources.SliceV6* attribute), 105
- data (*psd\_tools.psd.image\_resources.ThumbnailResource* attribute), 105
- data (*psd\_tools.psd.layer\_and\_mask.ChannelData* attribute), 112
- data (*psd\_tools.psd.linked\_layer.LinkedLayer* attribute), 113
- data (*psd\_tools.psd.patterns.Pattern* attribute), 113
- data (*psd\_tools.psd.patterns.VirtualMemoryArray* attribute), 114
- data (*psd\_tools.psd.tagged\_blocks.SmartObjectLayerData* attribute), 117
- data (*psd\_tools.psd.tagged\_blocks.TaggedBlock* attribute), 115
- DateCreated (*psd\_tools.terminology.Key* attribute), 156
- Datum (*psd\_tools.terminology.Key* attribute), 156
- DCS (*psd\_tools.terminology.Key* attribute), 156
- DCS (*psd\_tools.terminology.Type* attribute), 174
- DeepDepth (*psd\_tools.terminology.Type* attribute), 174
- DefinePattern (*psd\_tools.terminology.Event* attribute), 146
- Definition (*psd\_tools.terminology.Key* attribute), 156
- Defringe (*psd\_tools.terminology.Event* attribute), 146
- DeInterlace (*psd\_tools.terminology.Event* attribute), 146
- Delete (*psd\_tools.terminology.Event* attribute), 146
- density (*psd\_tools.api.adjustments.PhotoFilter* attribute), 32
- Density (*psd\_tools.terminology.Key* attribute), 156
- Density (*psd\_tools.terminology.Unit* attribute), 178
- depth (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- depth (*psd\_tools.psd.effects\_layer.BevelInfo* attribute), 95
- depth (*psd\_tools.psd.filter\_effects.FilterEffect* attribute), 97
- depth (*psd\_tools.psd.header.FileHeader* attribute), 98
- depth (*psd\_tools.psd.patterns.VirtualMemoryArray* attribute), 114
- depth (*psd\_tools.PSDImage* attribute), 13
- Depth (*psd\_tools.terminology.Key* attribute), 156
- Depth (*psd\_tools.terminology.Type* attribute), 174
- Desaturate (*psd\_tools.terminology.Enum* attribute), 130
- Desaturate (*psd\_tools.terminology.Event* attribute), 146
- descendants () (*psd\_tools.api.layers.Artboard* method), 42
- descendants () (*psd\_tools.api.layers.Group* method), 47
- descendants () (*psd\_tools.PSDImage* method), 13
- Descriptor (*class in psd\_tools.psd.descriptor*), 88
- Deselect (*psd\_tools.terminology.Event* attribute), 146
- Despeckle (*psd\_tools.terminology.Event* attribute), 146
- DestBlackMax (*psd\_tools.terminology.Key* attribute), 156
- DestBlackMin (*psd\_tools.terminology.Key* attribute), 156
- DestinationMode (*psd\_tools.terminology.Key* attribute), 156
- DestWhiteMax (*psd\_tools.terminology.Key* attribute), 156
- DestWhiteMin (*psd\_tools.terminology.Key* attribute), 156
- Detail (*psd\_tools.terminology.Key* attribute), 156
- Diameter (*psd\_tools.terminology.Key* attribute), 156
- Diamond (*psd\_tools.terminology.Enum* attribute), 130
- DicomFormat (*psd\_tools.terminology.Klass* attribute), 122
- Dict (*class in psd\_tools.psd.engine\_data*), 93
- DictElement (*class in psd\_tools.psd.base*), 86
- DIFFERENCE (*psd\_tools.constants.BlendMode* attribute), 73



- Difference (*psd\_tools.terminology.Enum* attribute), 130
- DifferenceClouds (*psd\_tools.terminology.Event* attribute), 146
- Diffuse (*psd\_tools.terminology.Event* attribute), 146
- DiffuseGlow (*psd\_tools.terminology.Event* attribute), 146
- DiffuseMode (*psd\_tools.terminology.Type* attribute), 175
- Diffusion (*psd\_tools.terminology.Enum* attribute), 130
- DiffusionDither (*psd\_tools.terminology.Enum* attribute), 130
- DiffusionDither (*psd\_tools.terminology.Key* attribute), 156
- direction (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- direction (*psd\_tools.psd.effects\_layer.BevelInfo* attribute), 96
- Direction (*psd\_tools.terminology.Key* attribute), 156
- Direction (*psd\_tools.terminology.Type* attribute), 175
- DirectionBalance (*psd\_tools.terminology.Key* attribute), 156
- disable (*psd\_tools.psd.vector.VectorMaskSetting* attribute), 120
- disabled (*psd\_tools.api.mask.Mask* attribute), 66
- disabled (*psd\_tools.api.shape.VectorMask* attribute), 67
- DisableLayerFX (*psd\_tools.terminology.Event* attribute), 146
- Displace (*psd\_tools.terminology.Event* attribute), 146
- DisplaceFile (*psd\_tools.terminology.Key* attribute), 156
- DisplacementMap (*psd\_tools.terminology.Key* attribute), 156
- DisplacementMap (*psd\_tools.terminology.Type* attribute), 175
- DISPLAY\_INFO (*psd\_tools.constants.Resource* attribute), 78
- DISPLAY\_INFO\_OBSOLETE (*psd\_tools.constants.Resource* attribute), 78
- DisplayCursorsPreferences (*psd\_tools.terminology.Enum* attribute), 130
- DisplayPrefs (*psd\_tools.terminology.Key* attribute), 156
- DisplayPrefs (*psd\_tools.terminology.Klass* attribute), 122
- DISSOLVE (*psd\_tools.constants.BlendMode* attribute), 73
- Dissolve (*psd\_tools.terminology.Enum* attribute), 130
- distance (*psd\_tools.api.effects.DropShadow* attribute), 34
- distance (*psd\_tools.api.effects.InnerShadow* attribute), 35
- distance (*psd\_tools.api.effects.Satin* attribute), 41
- distance (*psd\_tools.psd.effects\_layer.ShadowInfo* attribute), 94
- Distance (*psd\_tools.terminology.Key* attribute), 156
- Distance (*psd\_tools.terminology.Unit* attribute), 178
- Distort (*psd\_tools.terminology.Enum* attribute), 130
- Distortion (*psd\_tools.terminology.Key* attribute), 156
- Distribute (*psd\_tools.terminology.Event* attribute), 146
- Distribution (*psd\_tools.terminology.Key* attribute), 156
- Distribution (*psd\_tools.terminology.Type* attribute), 175
- Dither (*psd\_tools.terminology.Key* attribute), 156
- Dither (*psd\_tools.terminology.Type* attribute), 175
- DitherAmount (*psd\_tools.terminology.Key* attribute), 156
- dithered (*psd\_tools.api.adjustments.GradientMap* attribute), 33
- dithered (*psd\_tools.api.effects.GradientOverlay* attribute), 38
- DitherPreserve (*psd\_tools.terminology.Key* attribute), 156
- DitherQuality (*psd\_tools.terminology.Key* attribute), 157
- DitherQuality (*psd\_tools.terminology.Type* attribute), 175
- DIVIDE (*psd\_tools.constants.BlendMode* attribute), 73
- Document (*psd\_tools.terminology.Klass* attribute), 122
- document\_resources (*psd\_tools.api.layers.TypeLayer* attribute), 63
- DocumentID (*psd\_tools.terminology.Key* attribute), 157
- DocumentReference (*psd\_tools.terminology.Type* attribute), 175
- DodgeH (*psd\_tools.terminology.Enum* attribute), 130
- DodgeM (*psd\_tools.terminology.Enum* attribute), 130
- DodgeS (*psd\_tools.terminology.Enum* attribute), 131
- DodgeTool (*psd\_tools.terminology.Klass* attribute), 122
- DotGain (*psd\_tools.terminology.Key* attribute), 157
- DotGainCurves (*psd\_tools.terminology.Key* attribute), 157
- Dots (*psd\_tools.terminology.Enum* attribute), 131
- Double (*class in psd\_tools.psd.descriptor*), 88
- DPXFormat (*psd\_tools.terminology.Key* attribute), 156
- Draft (*psd\_tools.terminology.Enum* attribute), 131
- Draw (*psd\_tools.terminology.Event* attribute), 146
- DROP\_SHADOW (*psd\_tools.constants.EffectOSType* attribute), 76

- DropShadow (*class in psd\_tools.api.effects*), 34
- DropShadow (*psd\_tools.terminology.Key attribute*), 157
- DropShadow (*psd\_tools.terminology.Klass attribute*), 122
- DryBrush (*psd\_tools.terminology.Event attribute*), 146
- DUOTONE (*psd\_tools.constants.ColorMode attribute*), 75
- Duotone (*psd\_tools.terminology.Enum attribute*), 131
- DUOTONE\_HALFTONING\_INFO (*psd\_tools.constants.Resource attribute*), 78
- DUOTONE\_IMAGE\_INFO (*psd\_tools.constants.Resource attribute*), 78
- DUOTONE\_TRANSFER\_FUNCTION (*psd\_tools.constants.Resource attribute*), 78
- DuotoneInk (*psd\_tools.terminology.Klass attribute*), 122
- DuotoneMode (*psd\_tools.terminology.Klass attribute*), 122
- Duplicate (*psd\_tools.terminology.Event attribute*), 146
- Duplicate (*psd\_tools.terminology.Key attribute*), 157
- DustAndScratches (*psd\_tools.terminology.Event attribute*), 146
- DynamicColorSliders (*psd\_tools.terminology.Key attribute*), 157
- ## E
- EBUITU (*psd\_tools.terminology.Enum attribute*), 131
- Edge (*psd\_tools.terminology.Key attribute*), 157
- EdgeBrightness (*psd\_tools.terminology.Key attribute*), 157
- EdgeFidelity (*psd\_tools.terminology.Key attribute*), 157
- EdgeGlow (*psd\_tools.terminology.Enum attribute*), 131
- EdgeIntensity (*psd\_tools.terminology.Key attribute*), 157
- EdgeSimplicity (*psd\_tools.terminology.Key attribute*), 157
- EdgeThickness (*psd\_tools.terminology.Key attribute*), 157
- EdgeWidth (*psd\_tools.terminology.Key attribute*), 157
- Effect (*psd\_tools.terminology.Key attribute*), 157
- EFFECTIVE\_BW (*psd\_tools.constants.Resource attribute*), 78
- EffectOSType (*class in psd\_tools.constants*), 76
- Effects (*class in psd\_tools.api.effects*), 34
- effects (*psd\_tools.api.adjustments.GradientFill attribute*), 26
- effects (*psd\_tools.api.adjustments.PatternFill attribute*), 22
- effects (*psd\_tools.api.adjustments.SolidColorFill attribute*), 18
- effects (*psd\_tools.api.layers.Artboard attribute*), 43
- effects (*psd\_tools.api.layers.Group attribute*), 47
- effects (*psd\_tools.api.layers.PixelLayer attribute*), 51
- effects (*psd\_tools.api.layers.ShapeLayer attribute*), 55
- effects (*psd\_tools.api.layers.SmartObjectLayer attribute*), 59
- effects (*psd\_tools.api.layers.TypeLayer attribute*), 63
- EFFECTS\_LAYER (*psd\_tools.constants.Tag attribute*), 82
- EFFECTS\_VISIBLE (*psd\_tools.constants.Resource attribute*), 78
- EffectsLayer (*class in psd\_tools.psd.effects\_layer*), 94
- Element (*psd\_tools.terminology.Klass attribute*), 122
- ElementReference (*psd\_tools.terminology.Type attribute*), 175
- EliminateEvenFields (*psd\_tools.terminology.Enum attribute*), 131
- EliminateOddFields (*psd\_tools.terminology.Enum attribute*), 131
- Ellipse (*class in psd\_tools.api.shape*), 70
- Ellipse (*psd\_tools.terminology.Enum attribute*), 131
- Ellipse (*psd\_tools.terminology.Klass attribute*), 122
- EmbedCMYK (*psd\_tools.terminology.Key attribute*), 157
- EmbedGray (*psd\_tools.terminology.Key attribute*), 157
- EmbedLab (*psd\_tools.terminology.Key attribute*), 157
- EmbedProfiles (*psd\_tools.terminology.Key attribute*), 157
- EmbedRGB (*psd\_tools.terminology.Key attribute*), 157
- Emboss (*psd\_tools.terminology.Enum attribute*), 131
- Emboss (*psd\_tools.terminology.Event attribute*), 146
- EmptyElement (*class in psd\_tools.psd.base*), 85
- EmulsionDown (*psd\_tools.terminology.Key attribute*), 157
- enable\_colorization (*psd\_tools.api.adjustments.HueSaturation attribute*), 31
- enabled (*psd\_tools.api.effects.BevelEmboss attribute*), 40
- enabled (*psd\_tools.api.effects.ColorOverlay attribute*), 37
- enabled (*psd\_tools.api.effects.DropShadow attribute*), 34
- enabled (*psd\_tools.api.effects.Effects attribute*), 34
- enabled (*psd\_tools.api.effects.GradientOverlay attribute*), 38
- enabled (*psd\_tools.api.effects.InnerGlow attribute*), 36
- enabled (*psd\_tools.api.effects.InnerShadow attribute*), 35
- enabled (*psd\_tools.api.effects.OuterGlow attribute*),

- 36
- enabled (*psd\_tools.api.effects.PatternOverlay attribute*), 38
- enabled (*psd\_tools.api.effects.Satin attribute*), 41
- enabled (*psd\_tools.api.effects.Stroke attribute*), 39
- enabled (*psd\_tools.api.shape.Stroke attribute*), 68
- enabled (*psd\_tools.psd.effects\_layer.BevelInfo attribute*), 96
- enabled (*psd\_tools.psd.effects\_layer.InnerGlowInfo attribute*), 95
- enabled (*psd\_tools.psd.effects\_layer.OuterGlowInfo attribute*), 95
- enabled (*psd\_tools.psd.effects\_layer.ShadowInfo attribute*), 94
- enabled (*psd\_tools.psd.effects\_layer.SolidFillInfo attribute*), 96
- Enabled (*psd\_tools.terminology.Key attribute*), 157
- EnableGestures (*psd\_tools.terminology.Key attribute*), 157
- Encoding (*psd\_tools.terminology.Key attribute*), 157
- Encoding (*psd\_tools.terminology.Type attribute*), 175
- End (*psd\_tools.terminology.Key attribute*), 157
- EndArrowhead (*psd\_tools.terminology.Key attribute*), 157
- EndRamp (*psd\_tools.terminology.Key attribute*), 157
- EndSustain (*psd\_tools.terminology.Key attribute*), 157
- Engine (*psd\_tools.terminology.Key attribute*), 157
- engine\_dict (*psd\_tools.api.layers.TypeLayer attribute*), 63
- EngineData (*class in psd\_tools.psd.engine\_data*), 92
- EngineData2 (*class in psd\_tools.psd.engine\_data*), 92
- Enum (*class in psd\_tools.terminology*), 126
- enum (*psd\_tools.psd.descriptor.Enumerated attribute*), 88
- enum (*psd\_tools.psd.descriptor.EnumeratedReference attribute*), 89
- Enumerated (*class in psd\_tools.psd.descriptor*), 88
- Enumerated (*psd\_tools.terminology.Form attribute*), 151
- EnumeratedReference (*class in psd\_tools.psd.descriptor*), 89
- EPS\_OPTIONS (*psd\_tools.constants.Resource attribute*), 78
- EPSTIFFPreview (*psd\_tools.terminology.Klass attribute*), 122
- EPSPICTPreview (*psd\_tools.terminology.Klass attribute*), 122
- EPSPreview (*psd\_tools.terminology.Type attribute*), 175
- EPSTIFFPreview (*psd\_tools.terminology.Klass attribute*), 122
- Equalize (*psd\_tools.terminology.Event attribute*), 146
- EraserKind (*psd\_tools.terminology.Key attribute*), 157
- EraserKind (*psd\_tools.terminology.Type attribute*), 175
- EraserTool (*psd\_tools.terminology.Klass attribute*), 122
- EraseToHistory (*psd\_tools.terminology.Key attribute*), 157
- Event (*class in psd\_tools.terminology*), 144
- Exact (*psd\_tools.terminology.Enum attribute*), 131
- ExactPoints (*psd\_tools.terminology.Key attribute*), 157
- Exchange (*psd\_tools.terminology.Event attribute*), 146
- EXCLUSION (*psd\_tools.constants.BlendMode attribute*), 73
- Exclusion (*psd\_tools.terminology.Enum attribute*), 131
- EXIF\_DATA\_1 (*psd\_tools.constants.Resource attribute*), 78
- EXIF\_DATA\_3 (*psd\_tools.constants.Resource attribute*), 78
- Expand (*psd\_tools.terminology.Event attribute*), 146
- expansion (*psd\_tools.api.adjustments.GradientMap attribute*), 33
- Export (*psd\_tools.terminology.Event attribute*), 146
- Export (*psd\_tools.terminology.Key attribute*), 157
- Export (*psd\_tools.terminology.Klass attribute*), 122
- EXPORT\_SETTING1 (*psd\_tools.constants.Tag attribute*), 82
- EXPORT\_SETTING2 (*psd\_tools.constants.Tag attribute*), 82
- ExportClipboard (*psd\_tools.terminology.Key attribute*), 157
- Exposure (*class in psd\_tools.api.adjustments*), 30
- exposure (*psd\_tools.api.adjustments.Exposure attribute*), 30
- EXPOSURE (*psd\_tools.constants.Tag attribute*), 82
- Exposure (*psd\_tools.terminology.Key attribute*), 157
- EXRf (*psd\_tools.terminology.Klass attribute*), 122
- Extend (*psd\_tools.terminology.Key attribute*), 157
- ExtendedQuality (*psd\_tools.terminology.Key attribute*), 158
- Extension (*psd\_tools.terminology.Key attribute*), 158
- ExtensionsQuery (*psd\_tools.terminology.Key attribute*), 158
- EXTERNAL (*psd\_tools.constants.LinkedLayerType attribute*), 76
- extra (*psd\_tools.api.adjustments.Curves attribute*), 30
- extra (*psd\_tools.psd.filter\_effects.FilterEffect attribute*), 97
- extract\_bbox () (*psd\_tools.api.layers.Artboard static method*), 43
- extract\_bbox () (*psd\_tools.api.layers.Group static method*), 47
- Extrude (*psd\_tools.terminology.Event attribute*), 146

- ExtrudeDepth (*psd\_tools.terminology.Key attribute*), 158
- ExtrudeMaskIncomplete (*psd\_tools.terminology.Key attribute*), 158
- ExtrudeRandom (*psd\_tools.terminology.Key attribute*), 158
- ExtrudeRandom (*psd\_tools.terminology.Type attribute*), 175
- ExtrudeSize (*psd\_tools.terminology.Key attribute*), 158
- ExtrudeSolidFace (*psd\_tools.terminology.Key attribute*), 158
- ExtrudeType (*psd\_tools.terminology.Key attribute*), 158
- ExtrudeType (*psd\_tools.terminology.Type attribute*), 175
- EyeDropperSample (*psd\_tools.terminology.Key attribute*), 158
- EyeDropperSample (*psd\_tools.terminology.Type attribute*), 175
- ## F
- Facet (*psd\_tools.terminology.Event attribute*), 146
- Fade (*psd\_tools.terminology.Event attribute*), 146
- FadeoutSteps (*psd\_tools.terminology.Key attribute*), 158
- FadeTo (*psd\_tools.terminology.Key attribute*), 158
- Falloff (*psd\_tools.terminology.Key attribute*), 158
- Faster (*psd\_tools.terminology.Enum attribute*), 131
- Feather (*psd\_tools.terminology.Event attribute*), 146
- Feather (*psd\_tools.terminology.Key attribute*), 158
- FiberLength (*psd\_tools.terminology.Key attribute*), 158
- Fibers (*psd\_tools.terminology.Event attribute*), 146
- File (*psd\_tools.terminology.Enum attribute*), 131
- File (*psd\_tools.terminology.Key attribute*), 158
- file\_version (*psd\_tools.psd.image\_resources.VersionInfo attribute*), 106
- FileCreator (*psd\_tools.terminology.Key attribute*), 158
- FileHeader (*class in psd\_tools.psd.header*), 98
- FileInfo (*psd\_tools.terminology.Enum attribute*), 131
- FileInfo (*psd\_tools.terminology.Key attribute*), 158
- FileInfo (*psd\_tools.terminology.Klass attribute*), 122
- filename (*psd\_tools.api.smart\_object.SmartObject attribute*), 72
- filename (*psd\_tools.psd.linked\_layer.LinkedLayer attribute*), 113
- FileReference (*psd\_tools.terminology.Key attribute*), 158
- FileSavePrefs (*psd\_tools.terminology.Key attribute*), 158
- FileSavePrefs (*psd\_tools.terminology.Klass attribute*), 122
- filesize (*psd\_tools.api.smart\_object.SmartObject attribute*), 72
- filesize (*psd\_tools.psd.linked\_layer.LinkedLayer attribute*), 113
- FilesList (*psd\_tools.terminology.Key attribute*), 158
- filetype (*psd\_tools.api.smart\_object.SmartObject attribute*), 73
- filetype (*psd\_tools.psd.linked\_layer.LinkedLayer attribute*), 113
- FileType (*psd\_tools.terminology.Key attribute*), 158
- Fill (*psd\_tools.terminology.Event attribute*), 146
- Fill (*psd\_tools.terminology.Key attribute*), 158
- Fill (*psd\_tools.terminology.Type attribute*), 175
- fill\_enabled (*psd\_tools.api.shape.Stroke attribute*), 68
- fill\_type (*psd\_tools.api.effects.Stroke attribute*), 39
- FillBack (*psd\_tools.terminology.Enum attribute*), 131
- FillColor (*psd\_tools.terminology.Key attribute*), 158
- FillColor (*psd\_tools.terminology.Type attribute*), 175
- FillContents (*psd\_tools.terminology.Type attribute*), 175
- FillFlash (*psd\_tools.terminology.Klass attribute*), 123
- FillFore (*psd\_tools.terminology.Enum attribute*), 131
- FillInverse (*psd\_tools.terminology.Enum attribute*), 131
- FillMode (*psd\_tools.terminology.Type attribute*), 175
- FillNeutral (*psd\_tools.terminology.Key attribute*), 158
- FillSame (*psd\_tools.terminology.Enum attribute*), 131
- FilmGrain (*psd\_tools.terminology.Event attribute*), 146
- Filter (*psd\_tools.terminology.Event attribute*), 146
- FILTER\_EFFECTS1 (*psd\_tools.constants.Tag attribute*), 82
- FILTER\_EFFECTS2 (*psd\_tools.constants.Tag attribute*), 82
- FILTER\_EFFECTS3 (*psd\_tools.constants.Tag attribute*), 82
- FILTER\_MASK (*psd\_tools.constants.Tag attribute*), 82
- FilterEffect (*class in psd\_tools.psd.filter\_effects*), 97
- FilterEffectChannel (*class in psd\_tools.psd.filter\_effects*), 97
- FilterEffectExtra (*class in psd\_tools.psd.filter\_effects*), 97
- FilterEffects (*class in psd\_tools.psd.filter\_effects*), 96
- FilterLayerPersistentData (*psd\_tools.terminology.Key attribute*), 158
- FilterLayerRandomSeed (*psd\_tools.terminology.Key attribute*), 158
- FilterMask (*class in psd\_tools.psd.tagged\_blocks*), 116



- find() (*psd\_tools.api.effects.Effects* method), 34
- FindEdges (*psd\_tools.terminology.Event* attribute), 146
- FineDots (*psd\_tools.terminology.Enum* attribute), 131
- Fingerpainting (*psd\_tools.terminology.Key* attribute), 158
- First (*psd\_tools.terminology.Enum* attribute), 131
- FirstIdle (*psd\_tools.terminology.Enum* attribute), 131
- FitOnScreen (*psd\_tools.terminology.Enum* attribute), 131
- flag (*psd\_tools.psd.tagged\_blocks.UserMask* attribute), 118
- flags (*psd\_tools.api.mask.Mask* attribute), 66
- flags (*psd\_tools.psd.layer\_and\_mask.LayerRecord* attribute), 108
- flags (*psd\_tools.psd.layer\_and\_mask.MaskData* attribute), 110
- FlareCenter (*psd\_tools.terminology.Key* attribute), 158
- FlashPixFormat (*psd\_tools.terminology.Klass* attribute), 123
- Flatness (*psd\_tools.terminology.Key* attribute), 158
- Flatten (*psd\_tools.terminology.Key* attribute), 158
- FlattenImage (*psd\_tools.terminology.Event* attribute), 147
- Flip (*psd\_tools.terminology.Event* attribute), 147
- FlipVertical (*psd\_tools.terminology.Key* attribute), 158
- Float (*class in psd\_tools.psd.engine\_data*), 93
- fmt (*psd\_tools.psd.image\_resources.ThumbnailResource* attribute), 105
- Focus (*psd\_tools.terminology.Key* attribute), 159
- Folders (*psd\_tools.terminology.Key* attribute), 159
- FontDesignAxes (*psd\_tools.terminology.Key* attribute), 159
- FontDesignAxes (*psd\_tools.terminology.Klass* attribute), 123
- FontDesignAxesVectors (*psd\_tools.terminology.Key* attribute), 159
- FontName (*psd\_tools.terminology.Key* attribute), 159
- FontScript (*psd\_tools.terminology.Key* attribute), 159
- FontStyleName (*psd\_tools.terminology.Key* attribute), 159
- FontTechnology (*psd\_tools.terminology.Key* attribute), 159
- ForcedColors (*psd\_tools.terminology.Key* attribute), 159
- ForcedColors (*psd\_tools.terminology.Type* attribute), 175
- ForegroundColor (*psd\_tools.terminology.Enum* attribute), 131
- ForegroundColor (*psd\_tools.terminology.Key* attribute), 159
- attribute), 159
- ForegroundLevel (*psd\_tools.terminology.Key* attribute), 159
- FOREIGN\_EFFECT\_ID (*psd\_tools.constants.Tag* attribute), 82
- Form (*class in psd\_tools.terminology*), 151
- Format (*psd\_tools.terminology.Key* attribute), 159
- Format (*psd\_tools.terminology.Klass* attribute), 123
- Forward (*psd\_tools.terminology.Enum* attribute), 131
- Forward (*psd\_tools.terminology.Key* attribute), 159
- FPXCompress (*psd\_tools.terminology.Key* attribute), 158
- FPXCompress (*psd\_tools.terminology.Type* attribute), 175
- FPXCompressLossyJPEG (*psd\_tools.terminology.Enum* attribute), 131
- FPXCompressNone (*psd\_tools.terminology.Enum* attribute), 131
- FPXQuality (*psd\_tools.terminology.Key* attribute), 158
- FPXSize (*psd\_tools.terminology.Key* attribute), 158
- FPXView (*psd\_tools.terminology.Key* attribute), 158
- Fragment (*psd\_tools.terminology.Event* attribute), 147
- FRAMED\_GROUP (*psd\_tools.constants.Tag* attribute), 82
- FrameFill (*psd\_tools.terminology.Type* attribute), 175
- FrameFX (*psd\_tools.terminology.Key* attribute), 159
- FrameFX (*psd\_tools.terminology.Klass* attribute), 123
- FrameStyle (*psd\_tools.terminology.Type* attribute), 175
- FrameWidth (*psd\_tools.terminology.Key* attribute), 159
- FreeTransform (*psd\_tools.terminology.Enum* attribute), 131
- FreeTransformCenterState (*psd\_tools.terminology.Key* attribute), 159
- freq (*psd\_tools.psd.image\_resources.HalfToneScreen* attribute), 101
- Frequency (*psd\_tools.terminology.Key* attribute), 159
- Fresco (*psd\_tools.terminology.Event* attribute), 147
- From (*psd\_tools.terminology.Key* attribute), 159
- FromBuiltin (*psd\_tools.terminology.Key* attribute), 159
- frombytes() (*psd\_tools.psd.base.BaseElement* class method), 85
- FromMode (*psd\_tools.terminology.Key* attribute), 159
- frompil() (*psd\_tools.PSDImage* class method), 13
- Front (*psd\_tools.terminology.Enum* attribute), 131
- FullDocument (*psd\_tools.terminology.Enum* attribute), 131
- FullSize (*psd\_tools.terminology.Enum* attribute), 131
- FunctionKey (*psd\_tools.terminology.Key* attribute), 159
- Fuzziness (*psd\_tools.terminology.Key* attribute), 159

## G

- gamma (*psd\_tools.api.adjustments.Exposure attribute*), 30
- Gamma (*psd\_tools.terminology.Key attribute*), 160
- GamutWarning (*psd\_tools.terminology.Key attribute*), 160
- GaussianBlur (*psd\_tools.terminology.Event attribute*), 147
- GaussianDistribution (*psd\_tools.terminology.Enum attribute*), 132
- GCR (*psd\_tools.terminology.Key attribute*), 159
- GeneralPreferences (*psd\_tools.terminology.Enum attribute*), 132
- GeneralPrefs (*psd\_tools.terminology.Key attribute*), 160
- GeneralPrefs (*psd\_tools.terminology.Klass attribute*), 123
- Get (*psd\_tools.terminology.Event attribute*), 147
- get\_data() (*psd\_tools.psd.image\_data.ImageData method*), 98
- get\_data() (*psd\_tools.psd.image\_resources.ImageResources method*), 100
- get\_data() (*psd\_tools.psd.layer\_and\_mask.ChannelData method*), 112
- get\_data() (*psd\_tools.psd.patterns.VirtualMemoryArray method*), 114
- get\_name() (*psd\_tools.psd.descriptor.Enumerated method*), 88
- GIF89aExport (*psd\_tools.terminology.Klass attribute*), 123
- GIFColorFileColors (*psd\_tools.terminology.Enum attribute*), 131
- GIFColorFileColorTable (*psd\_tools.terminology.Enum attribute*), 131
- GIFColorFileMicrosoftPalette (*psd\_tools.terminology.Enum attribute*), 131
- GIFColorFileType (*psd\_tools.terminology.Key attribute*), 159
- GIFColorFileType (*psd\_tools.terminology.Type attribute*), 175
- GIFColorLimit (*psd\_tools.terminology.Key attribute*), 159
- GIFExportCaption (*psd\_tools.terminology.Key attribute*), 159
- GIFFormat (*psd\_tools.terminology.Klass attribute*), 123
- GIFMaskChannelIndex (*psd\_tools.terminology.Key attribute*), 159
- GIFMaskChannelInverted (*psd\_tools.terminology.Key attribute*), 159
- GIFPaletteAdaptive (*psd\_tools.terminology.Enum attribute*), 131
- GIFPaletteExact (*psd\_tools.terminology.Enum attribute*), 131
- GIFPaletteFile (*psd\_tools.terminology.Key attribute*), 159
- GIFPaletteOther (*psd\_tools.terminology.Enum attribute*), 132
- GIFPaletteSystem (*psd\_tools.terminology.Enum attribute*), 132
- GIFPaletteType (*psd\_tools.terminology.Key attribute*), 159
- GIFPaletteType (*psd\_tools.terminology.Type attribute*), 175
- GIFRequiredColorSpaceIndexed (*psd\_tools.terminology.Enum attribute*), 132
- GIFRequiredColorSpaceRGB (*psd\_tools.terminology.Enum attribute*), 132
- GIFRequiredColorSpaceType (*psd\_tools.terminology.Key attribute*), 159
- GIFRequiredColorSpaceType (*psd\_tools.terminology.Type attribute*), 175
- GIFRowOrderInterlaced (*psd\_tools.terminology.Enum attribute*), 132
- GIFRowOrderNormal (*psd\_tools.terminology.Enum attribute*), 132
- GIFRowOrderType (*psd\_tools.terminology.Key attribute*), 159
- GIFRowOrderType (*psd\_tools.terminology.Type attribute*), 175
- GIFTransparentColor (*psd\_tools.terminology.Key attribute*), 159
- GIFTransparentColorBlue (*psd\_tools.terminology.Key attribute*), 159
- GIFTransparentColorGreen (*psd\_tools.terminology.Key attribute*), 159
- GIFTransparentColorRed (*psd\_tools.terminology.Key attribute*), 159
- GIFUseBestMatch (*psd\_tools.terminology.Key attribute*), 160
- Glass (*psd\_tools.terminology.Event attribute*), 147
- GLOBAL\_ALTITUDE (*psd\_tools.constants.Resource attribute*), 78
- GLOBAL\_ANGLE (*psd\_tools.constants.Resource attribute*), 78
- global\_layer\_mask\_info (*psd\_tools.psd.layer\_and\_mask.LayerAndMaskInformation attribute*), 106
- GlobalAngle (*psd\_tools.terminology.Key attribute*), 160
- GlobalAngle (*psd\_tools.terminology.Klass attribute*), 123

- GlobalClass (*psd\_tools.terminology.Type* attribute), 175
- GlobalLayerMaskInfo (class in *psd\_tools.psd.layer\_and\_mask*), 107
- GlobalLayerMaskKind (class in *psd\_tools.constants*), 76
- GlobalLightingAngle (*psd\_tools.terminology.Key* attribute), 160
- GlobalObject (class in *psd\_tools.psd.descriptor*), 89
- GlobalObject (*psd\_tools.terminology.Type* attribute), 175
- Gloss (*psd\_tools.terminology.Key* attribute), 160
- glow\_source (*psd\_tools.api.effects.InnerGlow* attribute), 37
- glow\_type (*psd\_tools.api.effects.InnerGlow* attribute), 37
- glow\_type (*psd\_tools.api.effects.OuterGlow* attribute), 36
- GlowAmount (*psd\_tools.terminology.Key* attribute), 160
- GlowingEdges (*psd\_tools.terminology.Event* attribute), 147
- GlowTechnique (*psd\_tools.terminology.Key* attribute), 160
- Good (*psd\_tools.terminology.Enum* attribute), 132
- gradient (*psd\_tools.api.effects.GradientOverlay* attribute), 38
- gradient (*psd\_tools.api.effects.InnerGlow* attribute), 37
- gradient (*psd\_tools.api.effects.OuterGlow* attribute), 36
- gradient (*psd\_tools.api.effects.Stroke* attribute), 39
- Gradient (*psd\_tools.terminology.Event* attribute), 147
- Gradient (*psd\_tools.terminology.Key* attribute), 160
- Gradient (*psd\_tools.terminology.Klass* attribute), 123
- GRADIENT\_FILL\_SETTING (*psd\_tools.constants.Tag* attribute), 82
- gradient\_kind (*psd\_tools.api.adjustments.GradientFill* attribute), 26
- GRADIENT\_MAP (*psd\_tools.constants.Tag* attribute), 82
- gradient\_name (*psd\_tools.api.adjustments.GradientMap* attribute), 33
- GradientFill (class in *psd\_tools.api.adjustments*), 25
- GradientFill (*psd\_tools.terminology.Enum* attribute), 132
- GradientFill (*psd\_tools.terminology.Key* attribute), 160
- GradientFill (*psd\_tools.terminology.Klass* attribute), 123
- GradientForm (*psd\_tools.terminology.Type* attribute), 175
- GradientMap (class in *psd\_tools.api.adjustments*), 33
- GradientMap (*psd\_tools.terminology.Event* attribute), 147
- GradientMap (*psd\_tools.terminology.Klass* attribute), 123
- GradientOverlay (class in *psd\_tools.api.effects*), 37
- GradientTool (*psd\_tools.terminology.Klass* attribute), 123
- GradientType (*psd\_tools.terminology.Type* attribute), 175
- Grain (*psd\_tools.terminology.Event* attribute), 147
- Grain (*psd\_tools.terminology.Key* attribute), 160
- GrainClumped (*psd\_tools.terminology.Enum* attribute), 132
- GrainContrasty (*psd\_tools.terminology.Enum* attribute), 132
- GrainEnlarged (*psd\_tools.terminology.Enum* attribute), 132
- GrainHorizontal (*psd\_tools.terminology.Enum* attribute), 132
- Graininess (*psd\_tools.terminology.Key* attribute), 160
- GrainRegular (*psd\_tools.terminology.Enum* attribute), 132
- GrainSoft (*psd\_tools.terminology.Enum* attribute), 132
- GrainSpeckle (*psd\_tools.terminology.Enum* attribute), 132
- GrainSprinkles (*psd\_tools.terminology.Enum* attribute), 132
- GrainStippled (*psd\_tools.terminology.Enum* attribute), 132
- GrainType (*psd\_tools.terminology.Key* attribute), 160
- GrainType (*psd\_tools.terminology.Type* attribute), 175
- GrainVertical (*psd\_tools.terminology.Enum* attribute), 132
- GrainyDots (*psd\_tools.terminology.Enum* attribute), 132
- GraphicPen (*psd\_tools.terminology.Event* attribute), 147
- Graphics (*psd\_tools.terminology.Enum* attribute), 132
- Gray (*psd\_tools.terminology.Enum* attribute), 132
- Gray (*psd\_tools.terminology.Key* attribute), 160
- Gray16 (*psd\_tools.terminology.Enum* attribute), 132
- Gray18 (*psd\_tools.terminology.Enum* attribute), 132
- Gray22 (*psd\_tools.terminology.Enum* attribute), 132
- Gray50 (*psd\_tools.terminology.Enum* attribute), 132
- GrayBehavior (*psd\_tools.terminology.Key* attribute), 160
- GrayBehavior (*psd\_tools.terminology.Type* attribute), 175
- GRAYSCALE (*psd\_tools.constants.ColorMode* attribute), 75
- GRAYSCALE (*psd\_tools.constants.ColorSpaceID* attribute), 75
- GrayScale (*psd\_tools.terminology.Enum* attribute), 132

- 132
- Grayscale (*psd\_tools.terminology.Enum attribute*), 132
- Grayscale (*psd\_tools.terminology.Klass attribute*), 123
- GRAYSCALE\_HALFTONING\_INFO (*psd\_tools.constants.Resource attribute*), 78
- GRAYSCALE\_TRANSFER\_FUNCTION (*psd\_tools.constants.Resource attribute*), 78
- GrayscaleMode (*psd\_tools.terminology.Klass attribute*), 123
- GraySetup (*psd\_tools.terminology.Key attribute*), 160
- GraySetup (*psd\_tools.terminology.Klass attribute*), 123
- green (*psd\_tools.api.adjustments.BlackAndWhite attribute*), 32
- green (*psd\_tools.psd.image\_resources.SliceV6 attribute*), 105
- Green (*psd\_tools.terminology.Enum attribute*), 132
- Green (*psd\_tools.terminology.Key attribute*), 160
- GreenBlackPoint (*psd\_tools.terminology.Key attribute*), 160
- GreenGamma (*psd\_tools.terminology.Key attribute*), 160
- Greens (*psd\_tools.terminology.Enum attribute*), 132
- GreenWhitePoint (*psd\_tools.terminology.Key attribute*), 160
- GreenX (*psd\_tools.terminology.Key attribute*), 160
- GreenY (*psd\_tools.terminology.Key attribute*), 160
- GRID\_AND\_GUIDES\_INFO (*psd\_tools.constants.Resource attribute*), 78
- GridColor (*psd\_tools.terminology.Key attribute*), 160
- GridCustomColor (*psd\_tools.terminology.Key attribute*), 160
- GridGuidesInfo (*class in psd\_tools.psd.image\_resources*), 101
- GridMajor (*psd\_tools.terminology.Key attribute*), 160
- GridMinor (*psd\_tools.terminology.Key attribute*), 160
- GridStyle (*psd\_tools.terminology.Key attribute*), 160
- GridUnits (*psd\_tools.terminology.Key attribute*), 160
- Group (*class in psd\_tools.api.layers*), 46
- Group (*psd\_tools.terminology.Event attribute*), 147
- Group (*psd\_tools.terminology.Key attribute*), 160
- group\_id (*psd\_tools.psd.image\_resources.SliceV6 attribute*), 104
- GroutWidth (*psd\_tools.terminology.Key attribute*), 160
- Grow (*psd\_tools.terminology.Event attribute*), 147
- GrowSelection (*psd\_tools.terminology.Key attribute*), 160
- Guide (*psd\_tools.terminology.Klass attribute*), 123
- GuideGridColor (*psd\_tools.terminology.Type attribute*), 175
- GuideGridStyle (*psd\_tools.terminology.Type attribute*), 175
- Guides (*psd\_tools.terminology.Key attribute*), 160
- GuidesColor (*psd\_tools.terminology.Key attribute*), 160
- GuidesCustomColor (*psd\_tools.terminology.Key attribute*), 160
- GuidesGridPreferences (*psd\_tools.terminology.Enum attribute*), 132
- GuidesPrefs (*psd\_tools.terminology.Key attribute*), 160
- GuidesPrefs (*psd\_tools.terminology.Klass attribute*), 123
- GuidesStyle (*psd\_tools.terminology.Key attribute*), 161
- GutterWidth (*psd\_tools.terminology.Key attribute*), 161
- ## H
- HalftoneFile (*psd\_tools.terminology.Enum attribute*), 132
- HalftoneFile (*psd\_tools.terminology.Key attribute*), 161
- HalftoneScreen (*class in psd\_tools.psd.image\_resources*), 101
- HalftoneScreen (*psd\_tools.terminology.Enum attribute*), 133
- HalftoneScreen (*psd\_tools.terminology.Event attribute*), 147
- HalftoneScreen (*psd\_tools.terminology.Key attribute*), 161
- HalftoneScreen (*psd\_tools.terminology.Klass attribute*), 123
- HalftoneScreens (*class in psd\_tools.psd.image\_resources*), 101
- HalftoneSize (*psd\_tools.terminology.Key attribute*), 161
- HalftoneSpec (*psd\_tools.terminology.Key attribute*), 161
- HalftoneSpec (*psd\_tools.terminology.Klass attribute*), 123
- HARD\_LIGHT (*psd\_tools.constants.BlendMode attribute*), 74
- HARD\_MIX (*psd\_tools.constants.BlendMode attribute*), 74
- HardLight (*psd\_tools.terminology.Enum attribute*), 133
- Hardness (*psd\_tools.terminology.Key attribute*), 161
- has\_clip\_layers (*psd\_tools.api.adjustments.GradientFill method*), 26





- `method`), 19
- `has_stroke()` (`psd_tools.api.layers.Artboard` `method`), 43
- `has_stroke()` (`psd_tools.api.layers.Group` `method`), 47
- `has_stroke()` (`psd_tools.api.layers.PixelLayer` `method`), 51
- `has_stroke()` (`psd_tools.api.layers.ShapeLayer` `method`), 55
- `has_stroke()` (`psd_tools.api.layers.SmartObjectLayer` `method`), 59
- `has_stroke()` (`psd_tools.api.layers.TypeLayer` `method`), 63
- `has_thumbnail()` (`psd_tools.PSDImage` `method`), 13
- `has_vector_mask()` (`psd_tools.api.adjustments.GradientFill` `method`), 27
- `has_vector_mask()` (`psd_tools.api.adjustments.PatternFill` `method`), 22
- `has_vector_mask()` (`psd_tools.api.adjustments.SolidColorFill` `method`), 19
- `has_vector_mask()` (`psd_tools.api.layers.Artboard` `method`), 43
- `has_vector_mask()` (`psd_tools.api.layers.Group` `method`), 47
- `has_vector_mask()` (`psd_tools.api.layers.PixelLayer` `method`), 51
- `has_vector_mask()` (`psd_tools.api.layers.ShapeLayer` `method`), 55
- `has_vector_mask()` (`psd_tools.api.layers.SmartObjectLayer` `method`), 59
- `has_vector_mask()` (`psd_tools.api.layers.TypeLayer` `method`), 63
- `HasCmdHPreference` (`psd_tools.terminology.Key` `attribute`), 161
- `HDR_TONING_INFO` (`psd_tools.constants.Resource` `attribute`), 78
- `HDTV` (`psd_tools.terminology.Enum` `attribute`), 132
- `header` (`psd_tools.psd.PSD` `attribute`), 84
- `Header` (`psd_tools.terminology.Key` `attribute`), 161
- `Headline` (`psd_tools.terminology.Key` `attribute`), 161
- `Heavy` (`psd_tools.terminology.Enum` `attribute`), 133
- `height` (`psd_tools.api.adjustments.GradientFill` `attribute`), 27
- `height` (`psd_tools.api.adjustments.PatternFill` `attribute`), 23
- `height` (`psd_tools.api.adjustments.SolidColorFill` `attribute`), 19
- `height` (`psd_tools.api.layers.Artboard` `attribute`), 43
- `height` (`psd_tools.api.layers.Group` `attribute`), 47
- `height` (`psd_tools.api.layers.PixelLayer` `attribute`), 51
- `height` (`psd_tools.api.layers.ShapeLayer` `attribute`), 55
- `height` (`psd_tools.api.layers.SmartObjectLayer` `attribute`), 59
- `height` (`psd_tools.api.layers.TypeLayer` `attribute`), 63
- `height` (`psd_tools.api.mask.Mask` `attribute`), 66
- `height` (`psd_tools.psd.header.FileHeader` `attribute`), 98
- `height` (`psd_tools.psd.image_resources.ThumbnailResource` `attribute`), 105
- `height` (`psd_tools.psd.layer_and_mask.LayerRecord` `attribute`), 108
- `height` (`psd_tools.psd.layer_and_mask.MaskData` `attribute`), 110
- `height` (`psd_tools.PSDImage` `attribute`), 13
- `Height` (`psd_tools.terminology.Key` `attribute`), 161
- `height_unit` (`psd_tools.psd.image_resources.ResolutionInfo` `attribute`), 103
- `Hide` (`psd_tools.terminology.Event` `attribute`), 147
- `HideAll` (`psd_tools.terminology.Enum` `attribute`), 133
- `HideSelection` (`psd_tools.terminology.Enum` `attribute`), 133
- `High` (`psd_tools.terminology.Enum` `attribute`), 133
- `highlight_blend_mode` (`psd_tools.psd.effects_layer.BevelInfo` `attribute`), 96
- `highlight_color` (`psd_tools.api.effects.BevelEmboss` `attribute`), 40
- `highlight_color` (`psd_tools.psd.effects_layer.BevelInfo` `attribute`), 96
- `highlight_mode` (`psd_tools.api.effects.BevelEmboss` `attribute`), 40
- `highlight_opacity` (`psd_tools.api.effects.BevelEmboss` `attribute`), 40
- `highlight_opacity` (`psd_tools.psd.effects_layer.BevelInfo` `attribute`), 96
- `HighlightArea` (`psd_tools.terminology.Key` `attribute`), 161
- `HighlightColor` (`psd_tools.terminology.Key` `attribute`), 161
- `HighlightLevels` (`psd_tools.terminology.Key` `attribute`), 161
- `HighlightMode` (`psd_tools.terminology.Key` `attribute`), 161
- `HighlightOpacity` (`psd_tools.terminology.Key` `attribute`), 161
- `highlights` (`psd_tools.api.adjustments.ColorBalance` `attribute`), 31
- `Highlights` (`psd_tools.terminology.Enum` `attribute`), 133
- `HighlightStrength` (`psd_tools.terminology.Key` `attribute`), 161

- tribute), 161
  - HighPass (*psd\_tools.terminology.Event attribute*), 147
  - HighQuality (*psd\_tools.terminology.Enum attribute*), 133
  - Histogram (*psd\_tools.terminology.Enum attribute*), 133
  - History (*psd\_tools.terminology.Enum attribute*), 133
  - HistoryBrushSource (*psd\_tools.terminology.Key attribute*), 161
  - HistoryBrushTool (*psd\_tools.terminology.Klass attribute*), 123
  - HistoryPaletteOptions (*psd\_tools.terminology.Enum attribute*), 133
  - HistoryPreferences (*psd\_tools.terminology.Enum attribute*), 133
  - HistoryPrefs (*psd\_tools.terminology.Key attribute*), 161
  - HistoryPrefs (*psd\_tools.terminology.Klass attribute*), 123
  - HistoryState (*psd\_tools.terminology.Klass attribute*), 123
  - HistoryStates (*psd\_tools.terminology.Key attribute*), 161
  - HistoryStateSource (*psd\_tools.terminology.Key attribute*), 161
  - HistoryStateSource (*psd\_tools.terminology.Type attribute*), 175
  - horizontal (*psd\_tools.psd.image\_resources.ResolutionInfo attribute*), 103
  - horizontal (*psd\_tools.psd.image\_resources.SliceV6 attribute*), 104
  - Horizontal (*psd\_tools.terminology.Enum attribute*), 133
  - Horizontal (*psd\_tools.terminology.Key attribute*), 161
  - horizontal\_unit (*psd\_tools.psd.image\_resources.ResolutionInfo attribute*), 103
  - HorizontalLocation (*psd\_tools.terminology.Type attribute*), 175
  - HorizontalOnly (*psd\_tools.terminology.Enum attribute*), 133
  - HorizontalScale (*psd\_tools.terminology.Key attribute*), 161
  - HostName (*psd\_tools.terminology.Key attribute*), 161
  - HostVersion (*psd\_tools.terminology.Key attribute*), 161
  - HSB (*psd\_tools.constants.ColorSpaceID attribute*), 75
  - HSBColor (*psd\_tools.terminology.Enum attribute*), 132
  - HSBColor (*psd\_tools.terminology.Klass attribute*), 123
  - HSBColorMode (*psd\_tools.terminology.Klass attribute*), 123
  - HSBHSL (*psd\_tools.terminology.Event attribute*), 147
  - HSLColor (*psd\_tools.terminology.Enum attribute*), 132
  - HUE (*psd\_tools.constants.BlendMode attribute*), 74
  - Hue (*psd\_tools.terminology.Enum attribute*), 133
  - Hue (*psd\_tools.terminology.Key attribute*), 161
  - HUE\_SATURATION (*psd\_tools.constants.Tag attribute*), 82
  - HUE\_SATURATION\_V4 (*psd\_tools.constants.Tag attribute*), 82
  - HueSatAdjustment (*psd\_tools.terminology.Klass attribute*), 123
  - HueSatAdjustmentV2 (*psd\_tools.terminology.Klass attribute*), 123
  - HueSaturation (*class in psd\_tools.api.adjustments*), 31
  - HueSaturation (*psd\_tools.terminology.Event attribute*), 147
  - HueSaturation (*psd\_tools.terminology.Klass attribute*), 123
- ## I
- IBMPC (*psd\_tools.terminology.Enum attribute*), 133
  - ICC (*psd\_tools.terminology.Enum attribute*), 133
  - ICC\_PROFILE (*psd\_tools.constants.Resource attribute*), 78
  - ICC\_UNTAGGED\_PROFILE (*psd\_tools.constants.Resource attribute*), 78
  - ICCEngine (*psd\_tools.terminology.Key attribute*), 161
  - ICCSetupName (*psd\_tools.terminology.Key attribute*), 161
  - Icon (*psd\_tools.terminology.Enum attribute*), 133
  - id (*psd\_tools.psd.image\_resources.URLItem attribute*), 106
  - id (*psd\_tools.psd.layer\_and\_mask.ChannelInfo attribute*), 111
  - ID (*psd\_tools.terminology.Key attribute*), 161
  - Identifier (*class in psd\_tools.psd.descriptor*), 89
  - Identifier (*psd\_tools.terminology.Form attribute*), 151
  - Idle (*psd\_tools.terminology.Key attribute*), 161
  - IdleVM (*psd\_tools.terminology.Enum attribute*), 133
  - IDS\_SEED\_NUMBER (*psd\_tools.constants.Resource attribute*), 78
  - IFFFormat (*psd\_tools.terminology.Klass attribute*), 123
  - Ignore (*psd\_tools.terminology.Enum attribute*), 133
  - IllustratorPathsExport (*psd\_tools.terminology.Klass attribute*), 123
  - Image (*psd\_tools.terminology.Enum attribute*), 133
  - image\_data (*psd\_tools.psd.PSD attribute*), 84
  - image\_mode (*psd\_tools.psd.patterns.Pattern attribute*), 113
  - IMAGE\_MODE\_RAW (*psd\_tools.constants.Resource attribute*), 78

- IMAGE\_READY\_7\_ROLLOVER\_EXPANDED\_STATE  
(*psd\_tools.constants.Resource attribute*), 78
- IMAGE\_READY\_DATA\_SETS  
(*psd\_tools.constants.Resource attribute*), 78
- IMAGE\_READY\_DEFAULT\_SELECTED\_STATE  
(*psd\_tools.constants.Resource attribute*), 78
- IMAGE\_READY\_ROLLOVER\_EXPANDED\_STATE  
(*psd\_tools.constants.Resource attribute*), 78
- IMAGE\_READY\_SAVE\_LAYER\_SETTINGS  
(*psd\_tools.constants.Resource attribute*), 78
- IMAGE\_READY\_VARIABLES  
(*psd\_tools.constants.Resource attribute*), 78
- IMAGE\_READY\_VERSION  
(*psd\_tools.constants.Resource attribute*), 78
- image\_resources (*psd\_tools.psd.PSD attribute*), 84
- image\_resources (*psd\_tools.PSDImage attribute*), 13
- IMAGE\_STACK (*psd\_tools.constants.PlacedLayerType attribute*), 77
- ImageBalance (*psd\_tools.terminology.Key attribute*), 161
- ImageCachePreferences  
(*psd\_tools.terminology.Enum attribute*), 133
- ImageData (*class in psd\_tools.psd.image\_data*), 98
- ImagePoint (*psd\_tools.terminology.Klass attribute*), 123
- ImageReference (*psd\_tools.terminology.Type attribute*), 176
- ImageResource (*class in psd\_tools.psd.image\_resources*), 100
- ImageResources (*class in psd\_tools.psd.image\_resources*), 100
- ImageSize (*psd\_tools.terminology.Event attribute*), 147
- Import (*psd\_tools.terminology.Event attribute*), 147
- Import (*psd\_tools.terminology.Key attribute*), 161
- Import (*psd\_tools.terminology.Klass attribute*), 123
- Impressionist (*psd\_tools.terminology.Key attribute*), 161
- In (*psd\_tools.terminology.Key attribute*), 161
- Index (*class in psd\_tools.psd.descriptor*), 89
- index (*psd\_tools.api.shape.Ellipse attribute*), 70
- index (*psd\_tools.api.shape.Line attribute*), 70
- index (*psd\_tools.api.shape.Rectangle attribute*), 71
- index (*psd\_tools.api.shape.RoundedRectangle attribute*), 72
- index (*psd\_tools.psd.vector.Subpath attribute*), 119
- Index (*psd\_tools.terminology.Form attribute*), 151
- INDEXED (*psd\_tools.constants.ColorMode attribute*), 75
- INDEXED\_COLOR\_TABLE\_COUNT  
(*psd\_tools.constants.Resource attribute*), 78
- IndexedColor (*psd\_tools.terminology.Enum attribute*), 133
- IndexedColorMode (*psd\_tools.terminology.Klass attribute*), 123
- InfoPaletteOptions (*psd\_tools.terminology.Enum attribute*), 133
- InfoPaletteToggleSamplers  
(*psd\_tools.terminology.Enum attribute*), 133
- Inherits (*psd\_tools.terminology.Key attribute*), 161
- INITIAL\_FILL (*psd\_tools.constants.PathResourceID attribute*), 76
- initial\_fill\_rule  
(*psd\_tools.api.shape.VectorMask attribute*), 67
- InitialFillRule (*class in psd\_tools.psd.vector*), 120
- InkColors (*psd\_tools.terminology.Key attribute*), 161
- InkOutlines (*psd\_tools.terminology.Event attribute*), 147
- Inks (*psd\_tools.terminology.Key attribute*), 162
- InkTransfer (*psd\_tools.terminology.Klass attribute*), 123
- INNER\_GLOW (*psd\_tools.constants.EffectOSType attribute*), 76
- INNER\_SHADOW (*psd\_tools.constants.EffectOSType attribute*), 76
- InnerBevel (*psd\_tools.terminology.Enum attribute*), 133
- InnerGlow (*class in psd\_tools.api.effects*), 36
- InnerGlow (*psd\_tools.terminology.Key attribute*), 162
- InnerGlow (*psd\_tools.terminology.Klass attribute*), 123
- InnerGlowInfo (*class in psd\_tools.psd.effects\_layer*), 95
- InnerGlowSource (*psd\_tools.terminology.Key attribute*), 162
- InnerGlowSource (*psd\_tools.terminology.Type attribute*), 176
- InnerShadow (*class in psd\_tools.api.effects*), 35
- InnerShadow (*psd\_tools.terminology.Key attribute*), 162
- InnerShadow (*psd\_tools.terminology.Klass attribute*), 123
- Input (*psd\_tools.terminology.Key attribute*), 162
- InputBlackPoint (*psd\_tools.terminology.Key attribute*), 162
- InputMapRange (*psd\_tools.terminology.Key attribute*), 162
- InputRange (*psd\_tools.terminology.Key attribute*), 162
- InputWhitePoint (*psd\_tools.terminology.Key*

- attribute*), 162
- InsetFrame (*psd\_tools.terminology.Enum attribute*), 133
- Inside (*psd\_tools.terminology.Enum attribute*), 133
- Integer (*class in psd\_tools.psd.descriptor*), 89
- Integer (*class in psd\_tools.psd.engine\_data*), 93
- Integer (*class in psd\_tools.psd.image\_resources*), 102
- IntegerChannel (*psd\_tools.terminology.Type attribute*), 176
- IntegerElement (*class in psd\_tools.psd.base*), 85
- intensity (*psd\_tools.psd.effects\_layer.InnerGlowInfo attribute*), 95
- intensity (*psd\_tools.psd.effects\_layer.OuterGlowInfo attribute*), 95
- intensity (*psd\_tools.psd.effects\_layer.ShadowInfo attribute*), 94
- Intensity (*psd\_tools.terminology.Key attribute*), 162
- Intent (*psd\_tools.terminology.Key attribute*), 162
- Intent (*psd\_tools.terminology.Type attribute*), 176
- InterfaceBevelHighlight (*psd\_tools.terminology.Key attribute*), 162
- InterfaceBevelShadow (*psd\_tools.terminology.Key attribute*), 162
- InterfaceBlack (*psd\_tools.terminology.Key attribute*), 162
- InterfaceBorder (*psd\_tools.terminology.Key attribute*), 162
- InterfaceButtonDarkShadow (*psd\_tools.terminology.Key attribute*), 162
- InterfaceButtonDownFill (*psd\_tools.terminology.Key attribute*), 162
- InterfaceButtonUpFill (*psd\_tools.terminology.Key attribute*), 162
- InterfaceColor (*psd\_tools.terminology.Klass attribute*), 124
- InterfaceColorBlue2 (*psd\_tools.terminology.Key attribute*), 162
- InterfaceColorBlue32 (*psd\_tools.terminology.Key attribute*), 162
- InterfaceColorGreen2 (*psd\_tools.terminology.Key attribute*), 162
- InterfaceColorGreen32 (*psd\_tools.terminology.Key attribute*), 162
- InterfaceColorRed2 (*psd\_tools.terminology.Key attribute*), 162
- InterfaceColorRed32 (*psd\_tools.terminology.Key attribute*), 162
- InterfaceIconFillActive (*psd\_tools.terminology.Key attribute*), 162
- InterfaceIconFillDimmed (*psd\_tools.terminology.Key attribute*), 162
- InterfaceIconFillSelected (*psd\_tools.terminology.Key attribute*), 162
- InterfaceIconFrameActive (*psd\_tools.terminology.Key attribute*), 162
- InterfaceIconFrameDimmed (*psd\_tools.terminology.Key attribute*), 162
- InterfaceIconFrameSelected (*psd\_tools.terminology.Key attribute*), 162
- InterfacePaletteFill (*psd\_tools.terminology.Key attribute*), 162
- InterfaceRed (*psd\_tools.terminology.Key attribute*), 162
- InterfaceToolTipBackground (*psd\_tools.terminology.Key attribute*), 162
- InterfaceToolTipText (*psd\_tools.terminology.Key attribute*), 162
- InterfaceTransparencyBackground (*psd\_tools.terminology.Key attribute*), 162
- InterfaceTransparencyForeground (*psd\_tools.terminology.Key attribute*), 162
- InterfaceWhite (*psd\_tools.terminology.Key attribute*), 163
- Interlace (*psd\_tools.terminology.Key attribute*), 163
- InterlaceCreateType (*psd\_tools.terminology.Key attribute*), 163
- InterlaceCreateType (*psd\_tools.terminology.Type attribute*), 176
- InterlaceEliminateType (*psd\_tools.terminology.Key attribute*), 163
- InterlaceEliminateType (*psd\_tools.terminology.Type attribute*), 176
- interleave () (*psd\_tools.psd.color\_mode\_data.ColorModeData method*), 87
- interpolation (*psd\_tools.api.adjustments.GradientMap attribute*), 33
- Interpolation (*psd\_tools.terminology.Key attribute*), 163
- Interpolation (*psd\_tools.terminology.Type attribute*), 176
- InterpolationMethod (*psd\_tools.terminology.Key attribute*), 163
- Intersect (*psd\_tools.terminology.Event attribute*), 147
- IntersectWith (*psd\_tools.terminology.Event attribute*), 147
- Invalidated (*class in psd\_tools.api.shape*), 69
- invalidated (*psd\_tools.api.shape.Ellipse attribute*), 71
- invalidated (*psd\_tools.api.shape.Invalidated attribute*), 69
- invalidated (*psd\_tools.api.shape.Line attribute*), 70
- invalidated (*psd\_tools.api.shape.Rectangle attribute*), 71
- invalidated (*psd\_tools.api.shape.RoundedRectangle attribute*), 72
- Inverse (*psd\_tools.terminology.Event attribute*), 147
- INVERT (*psd\_tools.constants.Tag attribute*), 82



- invert (*psd\_tools.psd.effects\_layer.InnerGlowInfo* attribute), 95
  - invert (*psd\_tools.psd.vector.VectorMaskSetting* attribute), 120
  - Invert (*psd\_tools.terminology.Event* attribute), 147
  - Invert (*psd\_tools.terminology.Key* attribute), 163
  - Invert (*psd\_tools.terminology.Klass* attribute), 124
  - invert\_mask (*psd\_tools.psd.layer\_and\_mask.MaskFlags* attribute), 110
  - inverted (*psd\_tools.api.effects.Satin* attribute), 41
  - inverted (*psd\_tools.api.shape.VectorMask* attribute), 68
  - InvertMask (*psd\_tools.terminology.Key* attribute), 163
  - InvertSource2 (*psd\_tools.terminology.Key* attribute), 163
  - InvertTexture (*psd\_tools.terminology.Key* attribute), 163
  - IPTC\_NAA (*psd\_tools.constants.Resource* attribute), 78
  - is\_closed() (*psd\_tools.psd.vector.Subpath* method), 119
  - is\_group() (*psd\_tools.api.adjustments.GradientFill* method), 27
  - is\_group() (*psd\_tools.api.adjustments.PatternFill* method), 23
  - is\_group() (*psd\_tools.api.adjustments.SolidColorFill* method), 19
  - is\_group() (*psd\_tools.api.layers.Artboard* method), 43
  - is\_group() (*psd\_tools.api.layers.Group* method), 48
  - is\_group() (*psd\_tools.api.layers.PixelLayer* method), 51
  - is\_group() (*psd\_tools.api.layers.ShapeLayer* method), 55
  - is\_group() (*psd\_tools.api.layers.SmartObjectLayer* method), 59
  - is\_group() (*psd\_tools.api.layers.TypeLayer* method), 64
  - is\_group() (*psd\_tools.PSDImage* method), 14
  - is\_open (*psd\_tools.psd.tagged\_blocks.Annotation* attribute), 115
  - is\_path\_info (*psd\_tools.constants.Resource* attribute), 81
  - is\_plugin\_resource (*psd\_tools.constants.Resource* attribute), 81
  - is\_psd() (*psd\_tools.api.smart\_object.SmartObject* method), 73
  - is\_visible() (*psd\_tools.api.adjustments.GradientFill* method), 27
  - is\_visible() (*psd\_tools.api.adjustments.PatternFill* method), 23
  - is\_visible() (*psd\_tools.api.adjustments.SolidColorFill* method), 19
  - is\_visible() (*psd\_tools.api.layers.Artboard* method), 43
  - is\_visible() (*psd\_tools.api.layers.Group* method), 48
  - is\_visible() (*psd\_tools.api.layers.PixelLayer* method), 51
  - is\_visible() (*psd\_tools.api.layers.ShapeLayer* method), 55
  - is\_visible() (*psd\_tools.api.layers.SmartObjectLayer* method), 59
  - is\_visible() (*psd\_tools.api.layers.TypeLayer* method), 64
  - is\_visible() (*psd\_tools.PSDImage* method), 14
  - is\_written (*psd\_tools.psd.filter\_effects.FilterEffectChannel* attribute), 97
  - is\_written (*psd\_tools.psd.filter\_effects.FilterEffectExtra* attribute), 97
  - is\_written (*psd\_tools.psd.patterns.VirtualMemoryArray* attribute), 114
  - IsDirty (*psd\_tools.terminology.Key* attribute), 163
  - ItemIndex (*psd\_tools.terminology.Key* attribute), 163
  - items (*psd\_tools.psd.image\_resources.SlicesV6* attribute), 104
  - items\_count (*psd\_tools.psd.descriptor.ObjectArray* attribute), 90
- ## J
- JPEG (*psd\_tools.terminology.Enum* attribute), 133
  - JPEG\_QUALITY (*psd\_tools.constants.Resource* attribute), 79
  - JPEGFormat (*psd\_tools.terminology.Klass* attribute), 124
  - JPEGQuality (*psd\_tools.terminology.Key* attribute), 163
  - JUMP\_TO\_XPEP (*psd\_tools.constants.Resource* attribute), 79
  - JustifyAll (*psd\_tools.terminology.Enum* attribute), 133
  - JustifyFull (*psd\_tools.terminology.Enum* attribute), 133
- ## K
- KeepProfile (*psd\_tools.terminology.Enum* attribute), 133
  - Kelvin (*psd\_tools.terminology.Type* attribute), 176
  - KelvinCustomWhitePoint (*psd\_tools.terminology.Type* attribute), 176
  - Kerning (*psd\_tools.terminology.Key* attribute), 163
  - Key (*class in psd\_tools.terminology*), 151
  - key (*psd\_tools.psd.image\_resources.ImageResource* attribute), 100
  - key (*psd\_tools.psd.tagged\_blocks.TaggedBlock* attribute), 115

- key (*psd\_tools.psd.vector.VectorStrokeContentSetting* attribute), 120
- KeyboardPreferences (*psd\_tools.terminology.Enum* attribute), 133
- keyID (*psd\_tools.psd.descriptor.Property* attribute), 90
- Keywords (*psd\_tools.terminology.Key* attribute), 163
- kind (*psd\_tools.api.adjustments.GradientFill* attribute), 27
- kind (*psd\_tools.api.adjustments.PatternFill* attribute), 23
- kind (*psd\_tools.api.adjustments.SolidColorFill* attribute), 19
- kind (*psd\_tools.api.layers.Artboard* attribute), 44
- kind (*psd\_tools.api.layers.Group* attribute), 48
- kind (*psd\_tools.api.layers.PixelLayer* attribute), 51
- kind (*psd\_tools.api.layers.ShapeLayer* attribute), 55
- kind (*psd\_tools.api.layers.SmartObjectLayer* attribute), 59
- kind (*psd\_tools.api.layers.TypeLayer* attribute), 64
- kind (*psd\_tools.api.smart\_object.SmartObject* attribute), 73
- kind (*psd\_tools.psd.layer\_and\_mask.GlobalLayerMaskInfo* attribute), 107
- kind (*psd\_tools.psd.linked\_layer.LinkedLayer* attribute), 112
- kind (*psd\_tools.psd.tagged\_blocks.Annotation* attribute), 115
- kind (*psd\_tools.psd.tagged\_blocks.SectionDividerSetting* attribute), 117
- kind (*psd\_tools.psd.tagged\_blocks.SmartObjectLayerData* attribute), 117
- kind (*psd\_tools.PSDImage* attribute), 14
- Kind (*psd\_tools.terminology.Key* attribute), 163
- Klass (class in *psd\_tools.terminology*), 121
- KNOCKOUT\_SETTING (*psd\_tools.constants.Tag* attribute), 82
- Knot (class in *psd\_tools.psd.vector*), 119
- L**
- lab (*psd\_tools.api.adjustments.BrightnessContrast* attribute), 29
- LAB (*psd\_tools.constants.ColorMode* attribute), 75
- LAB (*psd\_tools.constants.ColorSpaceID* attribute), 75
- Lab (*psd\_tools.terminology.Enum* attribute), 133
- Lab48 (*psd\_tools.terminology.Enum* attribute), 133
- LabColor (*psd\_tools.terminology.Enum* attribute), 133
- LabColor (*psd\_tools.terminology.Klass* attribute), 124
- LabColorMode (*psd\_tools.terminology.Klass* attribute), 124
- Labels (*psd\_tools.terminology.Key* attribute), 163
- Landscape (*psd\_tools.terminology.Key* attribute), 163
- Large (*psd\_tools.terminology.Enum* attribute), 134
- LargeInteger (class in *psd\_tools.psd.descriptor*), 89
- Last (*psd\_tools.terminology.Enum* attribute), 134
- LastFilter (*psd\_tools.terminology.Enum* attribute), 134
- LastTransform (*psd\_tools.terminology.Key* attribute), 163
- LAYER (*psd\_tools.constants.Tag* attribute), 82
- Layer (*psd\_tools.terminology.Key* attribute), 163
- Layer (*psd\_tools.terminology.Klass* attribute), 124
- LAYER\_16 (*psd\_tools.constants.Tag* attribute), 82
- LAYER\_32 (*psd\_tools.constants.Tag* attribute), 82
- layer\_and\_mask\_information (*psd\_tools.psd.PSD* attribute), 84
- LAYER\_COMPS (*psd\_tools.constants.Resource* attribute), 79
- layer\_count (*psd\_tools.psd.layer\_and\_mask.LayerInfo* attribute), 107
- LAYER\_GROUP\_INFO (*psd\_tools.constants.Resource* attribute), 79
- LAYER\_GROUPS\_ENABLED\_ID (*psd\_tools.constants.Resource* attribute), 79
- layer\_id (*psd\_tools.api.adjustments.GradientFill* attribute), 27
- layer\_id (*psd\_tools.api.adjustments.PatternFill* attribute), 23
- layer\_id (*psd\_tools.api.adjustments.SolidColorFill* attribute), 19
- layer\_id (*psd\_tools.api.layers.Artboard* attribute), 44
- layer\_id (*psd\_tools.api.layers.Group* attribute), 48
- layer\_id (*psd\_tools.api.layers.PixelLayer* attribute), 52
- layer\_id (*psd\_tools.api.layers.ShapeLayer* attribute), 55
- layer\_id (*psd\_tools.api.layers.SmartObjectLayer* attribute), 59
- layer\_id (*psd\_tools.api.layers.TypeLayer* attribute), 64
- LAYER\_ID (*psd\_tools.constants.Tag* attribute), 82
- layer\_info (*psd\_tools.psd.layer\_and\_mask.LayerAndMaskInformation* attribute), 106
- layer\_knocks\_out (*psd\_tools.api.effects.DropShadow* attribute), 34
- LAYER\_MASK\_AS\_GLOBAL\_MASK (*psd\_tools.constants.Tag* attribute), 82
- LAYER\_NAME\_SOURCE\_SETTING (*psd\_tools.constants.Tag* attribute), 82
- layer\_records (*psd\_tools.psd.layer\_and\_mask.LayerInfo* attribute), 107
- LAYER\_SELECTION\_IDS (*psd\_tools.constants.Resource* attribute), 79
- LAYER\_STATE\_INFO (*psd\_tools.constants.Resource* attribute), 79
- LAYER\_VERSION (*psd\_tools.constants.Tag* attribute),

- 82
- LayerAndMaskInformation (class in *psd\_tools.psd.layer\_and\_mask*), 106
- LayerBlendingRanges (class in *psd\_tools.psd.layer\_and\_mask*), 109
- LayerEffects (*psd\_tools.terminology.Key* attribute), 163
- LayerEffects (*psd\_tools.terminology.Klass* attribute), 124
- LayerFlags (class in *psd\_tools.psd.layer\_and\_mask*), 109
- LayerFXVisible (*psd\_tools.terminology.Key* attribute), 163
- LayerFXVisible (*psd\_tools.terminology.Klass* attribute), 124
- LayerGroupEnabledIDs (class in *psd\_tools.psd.image\_resources*), 102
- LayerGroupInfo (class in *psd\_tools.psd.image\_resources*), 102
- LayerID (*psd\_tools.terminology.Key* attribute), 163
- LayerInfo (class in *psd\_tools.psd.layer\_and\_mask*), 107
- LayerName (*psd\_tools.terminology.Key* attribute), 163
- LayerOptions (*psd\_tools.terminology.Enum* attribute), 134
- LayerRecord (class in *psd\_tools.psd.layer\_and\_mask*), 108
- LayerRecords (class in *psd\_tools.psd.layer\_and\_mask*), 107
- Layers (*psd\_tools.terminology.Key* attribute), 163
- LayerSelectionIDs (class in *psd\_tools.psd.image\_resources*), 102
- LayersPaletteOptions (*psd\_tools.terminology.Enum* attribute), 134
- Leading (*psd\_tools.terminology.Key* attribute), 163
- leaving (*psd\_tools.psd.vector.Knot* attribute), 119
- left (*psd\_tools.api.adjustments.GradientFill* attribute), 27
- left (*psd\_tools.api.adjustments.PatternFill* attribute), 23
- left (*psd\_tools.api.adjustments.SolidColorFill* attribute), 19
- left (*psd\_tools.api.layers.Artboard* attribute), 44
- left (*psd\_tools.api.layers.PixelLayer* attribute), 52
- left (*psd\_tools.api.layers.ShapeLayer* attribute), 56
- left (*psd\_tools.api.layers.SmartObjectLayer* attribute), 60
- left (*psd\_tools.api.layers.TypeLayer* attribute), 64
- left (*psd\_tools.api.mask.Mask* attribute), 67
- left (*psd\_tools.psd.layer\_and\_mask.LayerRecord* attribute), 108
- left (*psd\_tools.psd.layer\_and\_mask.MaskData* attribute), 109
- left (*psd\_tools.psd.tagged\_blocks.TypeToolObjectSetting* attribute), 118
- left (*psd\_tools.psd.vector.ClipboardRecord* attribute), 119
- left (*psd\_tools.PSDImage* attribute), 14
- Left (*psd\_tools.terminology.Enum* attribute), 134
- Left (*psd\_tools.terminology.Key* attribute), 163
- Left\_PLUGIN (*psd\_tools.terminology.Enum* attribute), 134
- LegacySerialString (*psd\_tools.terminology.Key* attribute), 163
- length (*psd\_tools.api.adjustments.GradientMap* attribute), 33
- length (*psd\_tools.psd.layer\_and\_mask.ChannelInfo* attribute), 111
- Length (*psd\_tools.terminology.Key* attribute), 163
- Lens (*psd\_tools.terminology.Key* attribute), 163
- Lens (*psd\_tools.terminology.Type* attribute), 176
- LensFlare (*psd\_tools.terminology.Event* attribute), 147
- Level (*psd\_tools.terminology.Key* attribute), 163
- LevelBased (*psd\_tools.terminology.Enum* attribute), 134
- Levels (class in *psd\_tools.api.adjustments*), 30
- LEVELS (*psd\_tools.constants.Tag* attribute), 82
- Levels (*psd\_tools.terminology.Event* attribute), 147
- Levels (*psd\_tools.terminology.Key* attribute), 163
- Levels (*psd\_tools.terminology.Klass* attribute), 124
- LevelsAdjustment (*psd\_tools.terminology.Klass* attribute), 124
- Light (*psd\_tools.terminology.Enum* attribute), 134
- LightBlue (*psd\_tools.terminology.Enum* attribute), 134
- LightDark (*psd\_tools.terminology.Key* attribute), 163
- LightDirBottom (*psd\_tools.terminology.Enum* attribute), 134
- LightDirBottomLeft (*psd\_tools.terminology.Enum* attribute), 134
- LightDirBottomRight (*psd\_tools.terminology.Enum* attribute), 134
- LightDirection (*psd\_tools.terminology.Key* attribute), 163
- LightDirection (*psd\_tools.terminology.Type* attribute), 176
- LightDirectional (*psd\_tools.terminology.Enum* attribute), 134
- LightDirLeft (*psd\_tools.terminology.Enum* attribute), 134
- LightDirRight (*psd\_tools.terminology.Enum* attribute), 134
- LightDirTop (*psd\_tools.terminology.Enum* attribute), 134
- LightDirTopLeft (*psd\_tools.terminology.Enum* attribute), 134



- tribute*), 134
- LightDirTopRight (*psd\_tools.terminology.Enum attribute*), 134
- LIGHTEN (*psd\_tools.constants.BlendMode attribute*), 74
- Lighten (*psd\_tools.terminology.Enum attribute*), 134
- LightenGrout (*psd\_tools.terminology.Key attribute*), 164
- LightenOnly (*psd\_tools.terminology.Enum attribute*), 134
- LIGHTER\_COLOR (*psd\_tools.constants.BlendMode attribute*), 74
- LightGray (*psd\_tools.terminology.Enum attribute*), 134
- LightingEffects (*psd\_tools.terminology.Event attribute*), 147
- LightIntensity (*psd\_tools.terminology.Key attribute*), 164
- Lightness (*psd\_tools.terminology.Enum attribute*), 134
- Lightness (*psd\_tools.terminology.Key attribute*), 164
- LightOmni (*psd\_tools.terminology.Enum attribute*), 134
- LightPosBottom (*psd\_tools.terminology.Enum attribute*), 134
- LightPosBottomLeft (*psd\_tools.terminology.Enum attribute*), 134
- LightPosBottomRight (*psd\_tools.terminology.Enum attribute*), 134
- LightPosition (*psd\_tools.terminology.Key attribute*), 164
- LightPosition (*psd\_tools.terminology.Type attribute*), 176
- LightPosLeft (*psd\_tools.terminology.Enum attribute*), 134
- LightPosRight (*psd\_tools.terminology.Enum attribute*), 134
- LightPosTop (*psd\_tools.terminology.Enum attribute*), 134
- LightPosTopLeft (*psd\_tools.terminology.Enum attribute*), 134
- LightPosTopRight (*psd\_tools.terminology.Enum attribute*), 134
- LightRed (*psd\_tools.terminology.Enum attribute*), 134
- LIGHTROOM\_WORKFLOW (*psd\_tools.constants.Resource attribute*), 79
- LightSource (*psd\_tools.terminology.Key attribute*), 164
- LightSource (*psd\_tools.terminology.Klass attribute*), 124
- LightSpot (*psd\_tools.terminology.Enum attribute*), 134
- LightType (*psd\_tools.terminology.Key attribute*), 164
- LightType (*psd\_tools.terminology.Type attribute*), 176
- Line (*class in psd\_tools.api.shape*), 69
- Line (*psd\_tools.terminology.Enum attribute*), 134
- Line (*psd\_tools.terminology.Key attribute*), 164
- Line (*psd\_tools.terminology.Klass attribute*), 124
- line\_alignment (*psd\_tools.api.shape.Stroke attribute*), 68
- line\_cap\_type (*psd\_tools.api.shape.Stroke attribute*), 68
- line\_dash\_offset (*psd\_tools.api.shape.Stroke attribute*), 68
- line\_dash\_set (*psd\_tools.api.shape.Stroke attribute*), 68
- line\_end (*psd\_tools.api.shape.Line attribute*), 70
- line\_join\_type (*psd\_tools.api.shape.Stroke attribute*), 69
- line\_start (*psd\_tools.api.shape.Line attribute*), 70
- line\_weight (*psd\_tools.api.shape.Line attribute*), 70
- line\_width (*psd\_tools.api.shape.Stroke attribute*), 69
- Linear (*psd\_tools.terminology.Enum attribute*), 134
- LINEAR\_BURN (*psd\_tools.constants.BlendMode attribute*), 74
- LINEAR\_DODGE (*psd\_tools.constants.BlendMode attribute*), 74
- LINEAR\_LIGHT (*psd\_tools.constants.BlendMode attribute*), 74
- Lines (*psd\_tools.terminology.Enum attribute*), 135
- Link (*psd\_tools.terminology.Event attribute*), 147
- Linked (*psd\_tools.terminology.Enum attribute*), 135
- linked\_file (*psd\_tools.psd.linked\_layer.LinkedLayer attribute*), 113
- LINKED\_LAYER1 (*psd\_tools.constants.Tag attribute*), 82
- LINKED\_LAYER2 (*psd\_tools.constants.Tag attribute*), 82
- LINKED\_LAYER3 (*psd\_tools.constants.Tag attribute*), 82
- LINKED\_LAYER\_EXTERNAL (*psd\_tools.constants.Tag attribute*), 82
- LinkedLayer (*class in psd\_tools.psd.linked\_layer*), 112
- LinkedLayerIDs (*psd\_tools.terminology.Key attribute*), 164
- LinkedLayers (*class in psd\_tools.psd.linked\_layer*), 112
- LinkedLayerType (*class in psd\_tools.constants*), 76
- LinkEnable (*psd\_tools.terminology.Key attribute*), 164
- List (*class in psd\_tools.psd.descriptor*), 90
- List (*class in psd\_tools.psd.engine\_data*), 93
- ListElement (*class in psd\_tools.psd.base*), 86
- LocalLightingAltitude (*psd\_tools.terminology.Key attribute*), 164
- LocalLightingAngle (*psd\_tools.terminology.Key*

- attribute*), 164
  - LocalRange (*psd\_tools.terminology.Key attribute*), 164
  - Location (*psd\_tools.terminology.Key attribute*), 164
  - LocationReference (*psd\_tools.terminology.Type attribute*), 176
  - lock\_state (*psd\_tools.psd.linked\_layer.LinkedLayer attribute*), 113
  - Log (*psd\_tools.terminology.Key attribute*), 164
  - Logarithmic (*psd\_tools.terminology.Key attribute*), 164
  - LongLines (*psd\_tools.terminology.Enum attribute*), 135
  - LongStrokes (*psd\_tools.terminology.Enum attribute*), 135
  - Low (*psd\_tools.terminology.Enum attribute*), 135
  - Lower (*psd\_tools.terminology.Enum attribute*), 135
  - LowerCase (*psd\_tools.terminology.Key attribute*), 164
  - LowQuality (*psd\_tools.terminology.Enum attribute*), 135
  - Luminance (*psd\_tools.terminology.Key attribute*), 164
  - luminosity (*psd\_tools.api.adjustments.ColorBalance attribute*), 31
  - luminosity (*psd\_tools.api.adjustments.PhotoFilter attribute*), 32
  - LUMINOSITY (*psd\_tools.constants.BlendMode attribute*), 74
  - Luminosity (*psd\_tools.terminology.Enum attribute*), 135
  - LUTAnimation (*psd\_tools.terminology.Key attribute*), 163
  - LZWCompression (*psd\_tools.terminology.Key attribute*), 163
- ## M
- MAC\_NSPRINTFINFO (*psd\_tools.constants.Resource attribute*), 79
  - MAC\_PAGE\_FORMAT\_INFO (*psd\_tools.constants.Resource attribute*), 79
  - MAC\_PRINT\_MANAGER\_INFO (*psd\_tools.constants.Resource attribute*), 79
  - Macintosh (*psd\_tools.terminology.Enum attribute*), 135
  - MacintoshSystem (*psd\_tools.terminology.Enum attribute*), 135
  - MacPaintFormat (*psd\_tools.terminology.Klass attribute*), 124
  - MacThumbnail (*psd\_tools.terminology.Enum attribute*), 135
  - magenta (*psd\_tools.api.adjustments.BlackAndWhite attribute*), 32
  - Magenta (*psd\_tools.terminology.Enum attribute*), 135
  - Magenta (*psd\_tools.terminology.Key attribute*), 164
  - Magentas (*psd\_tools.terminology.Enum attribute*), 135
  - MagicEraserTool (*psd\_tools.terminology.Klass attribute*), 124
  - MagicPoint (*psd\_tools.terminology.Klass attribute*), 124
  - major\_version (*psd\_tools.psd.tagged\_blocks.Annotations attribute*), 115
  - Make (*psd\_tools.terminology.Event attribute*), 147
  - MakeVisible (*psd\_tools.terminology.Key attribute*), 164
  - ManipulationFOV (*psd\_tools.terminology.Key attribute*), 164
  - MapBlack (*psd\_tools.terminology.Key attribute*), 164
  - Mapping (*psd\_tools.terminology.Key attribute*), 164
  - MappingShape (*psd\_tools.terminology.Key attribute*), 164
  - Mask (*class in psd\_tools.api.mask*), 66
  - mask (*psd\_tools.api.adjustments.GradientFill attribute*), 27
  - mask (*psd\_tools.api.adjustments.PatternFill attribute*), 23
  - mask (*psd\_tools.api.adjustments.SolidColorFill attribute*), 19
  - mask (*psd\_tools.api.layers.Artboard attribute*), 44
  - mask (*psd\_tools.api.layers.Group attribute*), 48
  - mask (*psd\_tools.api.layers.PixelLayer attribute*), 52
  - mask (*psd\_tools.api.layers.ShapeLayer attribute*), 56
  - mask (*psd\_tools.api.layers.SmartObjectLayer attribute*), 60
  - mask (*psd\_tools.api.layers.TypeLayer attribute*), 64
  - Mask (*psd\_tools.terminology.Enum attribute*), 135
  - Mask (*psd\_tools.terminology.Klass attribute*), 124
  - mask\_data (*psd\_tools.psd.layer\_and\_mask.LayerRecord attribute*), 108
  - mask\_disabled (*psd\_tools.psd.layer\_and\_mask.MaskFlags attribute*), 110
  - MaskData (*class in psd\_tools.psd.layer\_and\_mask*), 109
  - MaskedAreas (*psd\_tools.terminology.Enum attribute*), 135
  - MaskFlags (*class in psd\_tools.psd.layer\_and\_mask*), 110
  - MaskIndicator (*psd\_tools.terminology.Type attribute*), 176
  - MaskParameters (*class in psd\_tools.psd.layer\_and\_mask*), 111
  - master (*psd\_tools.api.adjustments.HueSaturation attribute*), 31
  - master (*psd\_tools.api.adjustments.Levels attribute*), 30
  - MasterAdaptive (*psd\_tools.terminology.Enum attribute*), 135
  - MasterPerceptual (*psd\_tools.terminology.Enum attribute*), 135

- MasterSelective (*psd\_tools.terminology.Enum attribute*), 135
- Material (*psd\_tools.terminology.Key attribute*), 164
- Matrix (*psd\_tools.terminology.Key attribute*), 164
- MatteColor (*psd\_tools.terminology.Key attribute*), 164
- MatteColor (*psd\_tools.terminology.Type attribute*), 176
- MatteTechnique (*psd\_tools.terminology.Type attribute*), 176
- max\_channels (*psd\_tools.psd.filter\_effects.FilterEffect attribute*), 97
- max\_color (*psd\_tools.api.adjustments.GradientMap attribute*), 33
- Maximum (*psd\_tools.terminology.Enum attribute*), 135
- Maximum (*psd\_tools.terminology.Event attribute*), 147
- Maximum (*psd\_tools.terminology.Key attribute*), 164
- MaximumQuality (*psd\_tools.terminology.Enum attribute*), 135
- MaximumStates (*psd\_tools.terminology.Key attribute*), 164
- Maya (*psd\_tools.terminology.Enum attribute*), 135
- mean (*psd\_tools.api.adjustments.BrightnessContrast attribute*), 29
- MEASUREMENT\_SCALE (*psd\_tools.constants.Resource attribute*), 79
- Median (*psd\_tools.terminology.Event attribute*), 147
- Medium (*psd\_tools.terminology.Enum attribute*), 135
- MediumBlue (*psd\_tools.terminology.Enum attribute*), 135
- MediumDots (*psd\_tools.terminology.Enum attribute*), 135
- MediumLines (*psd\_tools.terminology.Enum attribute*), 135
- MediumQuality (*psd\_tools.terminology.Enum attribute*), 135
- MediumStrokes (*psd\_tools.terminology.Enum attribute*), 135
- MemoryPreferences (*psd\_tools.terminology.Enum attribute*), 135
- MemoryUsagePercent (*psd\_tools.terminology.Key attribute*), 164
- MenuItem (*psd\_tools.terminology.Klass attribute*), 124
- MenuItem (*psd\_tools.terminology.Type attribute*), 176
- Merge (*psd\_tools.terminology.Key attribute*), 164
- MergeChannels (*psd\_tools.terminology.Enum attribute*), 135
- Merged (*psd\_tools.terminology.Enum attribute*), 135
- Merged (*psd\_tools.terminology.Key attribute*), 164
- MergedLayers (*psd\_tools.terminology.Enum attribute*), 135
- MergedLayersOld (*psd\_tools.terminology.Enum attribute*), 135
- MergeLayers (*psd\_tools.terminology.Event attribute*), 147
- MergeLayersOld (*psd\_tools.terminology.Event attribute*), 147
- MergeSpotChannel (*psd\_tools.terminology.Event attribute*), 147
- MergeVisible (*psd\_tools.terminology.Event attribute*), 148
- message (*psd\_tools.psd.image\_resources.SliceV6 attribute*), 104
- Message (*psd\_tools.terminology.Key attribute*), 164
- METADATA\_SETTING (*psd\_tools.constants.Tag attribute*), 82
- MetadataSetting (*class in psd\_tools.psd.tagged\_blocks*), 116
- MetadataSettings (*class in psd\_tools.psd.tagged\_blocks*), 116
- method (*psd\_tools.api.adjustments.SelectiveColor attribute*), 33
- Method (*psd\_tools.terminology.Key attribute*), 164
- Method (*psd\_tools.terminology.Type attribute*), 176
- Mezzotint (*psd\_tools.terminology.Event attribute*), 148
- MezzotintType (*psd\_tools.terminology.Key attribute*), 164
- MezzotintType (*psd\_tools.terminology.Type attribute*), 176
- Middle (*psd\_tools.terminology.Enum attribute*), 135
- Midpoint (*psd\_tools.terminology.Key attribute*), 164
- MidtoneLevels (*psd\_tools.terminology.Key attribute*), 164
- midtone (*psd\_tools.api.adjustments.ColorBalance attribute*), 31
- Midtones (*psd\_tools.terminology.Enum attribute*), 135
- Millimeters (*psd\_tools.terminology.Unit attribute*), 178
- min\_color (*psd\_tools.api.adjustments.GradientMap attribute*), 33
- Minimum (*psd\_tools.terminology.Event attribute*), 148
- Minimum (*psd\_tools.terminology.Key attribute*), 165
- minor\_version (*psd\_tools.psd.tagged\_blocks.Annotations attribute*), 115
- MismatchCMYK (*psd\_tools.terminology.Key attribute*), 165
- MismatchGray (*psd\_tools.terminology.Key attribute*), 165
- MismatchRGB (*psd\_tools.terminology.Key attribute*), 165
- miter\_limit (*psd\_tools.api.shape.Stroke attribute*), 69
- mod\_time (*psd\_tools.psd.linked\_layer.LinkedLayer attribute*), 113
- mode (*psd\_tools.api.adjustments.GradientMap attribute*), 33
- Mode (*psd\_tools.terminology.Key attribute*), 165

- Mode (*psd\_tools.terminology.Klass* attribute), 124
- Mode (*psd\_tools.terminology.Type* attribute), 176
- ModeGray (*psd\_tools.terminology.Enum* attribute), 135
- ModeRGB (*psd\_tools.terminology.Enum* attribute), 135
- Monitor (*psd\_tools.terminology.Enum* attribute), 136
- MonitorSetup (*psd\_tools.terminology.Enum* attribute), 136
- Monochromatic (*psd\_tools.terminology.Key* attribute), 165
- monochrome (*psd\_tools.api.adjustments.ChannelMixer* attribute), 32
- Monotone (*psd\_tools.terminology.Enum* attribute), 136
- Mosaic (*psd\_tools.terminology.Event* attribute), 148
- Mosaic\_PLUGIN (*psd\_tools.terminology.Event* attribute), 148
- MotionBlur (*psd\_tools.terminology.Event* attribute), 148
- Move (*psd\_tools.terminology.Event* attribute), 148
- MoveTo (*psd\_tools.terminology.Key* attribute), 165
- Multi72Color (*psd\_tools.terminology.Enum* attribute), 136
- Multi72Gray (*psd\_tools.terminology.Enum* attribute), 136
- MULTICHANNEL (*psd\_tools.constants.ColorMode* attribute), 75
- Multichannel (*psd\_tools.terminology.Enum* attribute), 136
- MultichannelMode (*psd\_tools.terminology.Klass* attribute), 124
- MultiNoCompositePS (*psd\_tools.terminology.Enum* attribute), 136
- MULTIPLY (*psd\_tools.constants.BlendMode* attribute), 74
- Multiply (*psd\_tools.terminology.Enum* attribute), 136
- ## N
- Name (*class in psd\_tools.psd.descriptor*), 90
- name (*psd\_tools.api.adjustments.GradientFill* attribute), 27
- name (*psd\_tools.api.adjustments.PatternFill* attribute), 23
- name (*psd\_tools.api.adjustments.SolidColorFill* attribute), 19
- name (*psd\_tools.api.layers.Artboard* attribute), 44
- name (*psd\_tools.api.layers.Group* attribute), 48
- name (*psd\_tools.api.layers.PixelLayer* attribute), 52
- name (*psd\_tools.api.layers.ShapeLayer* attribute), 56
- name (*psd\_tools.api.layers.SmartObjectLayer* attribute), 60
- name (*psd\_tools.api.layers.TypeLayer* attribute), 64
- name (*psd\_tools.psd.descriptor.Class* attribute), 87
- name (*psd\_tools.psd.descriptor.Descriptor* attribute), 88
- name (*psd\_tools.psd.descriptor.EnumeratedReference* attribute), 89
- name (*psd\_tools.psd.descriptor.Name* attribute), 90
- name (*psd\_tools.psd.descriptor.ObjectArray* attribute), 90
- name (*psd\_tools.psd.descriptor.Offset* attribute), 91
- name (*psd\_tools.psd.descriptor.Property* attribute), 90
- name (*psd\_tools.psd.image\_resources.ImageResource* attribute), 101
- name (*psd\_tools.psd.image\_resources.SlicesV6* attribute), 104
- name (*psd\_tools.psd.image\_resources.SliceV6* attribute), 104
- name (*psd\_tools.psd.image\_resources.URLItem* attribute), 106
- name (*psd\_tools.psd.layer\_and\_mask.LayerRecord* attribute), 108
- name (*psd\_tools.psd.patterns.Pattern* attribute), 113
- name (*psd\_tools.PSDImage* attribute), 14
- Name (*psd\_tools.terminology.Key* attribute), 165
- native\_color (*psd\_tools.psd.effects\_layer.InnerGlowInfo* attribute), 95
- native\_color (*psd\_tools.psd.effects\_layer.OuterGlowInfo* attribute), 95
- native\_color (*psd\_tools.psd.effects\_layer.ShadowInfo* attribute), 94
- native\_color (*psd\_tools.psd.effects\_layer.SolidFillInfo* attribute), 96
- NavigatorPaletteOptions (*psd\_tools.terminology.Enum* attribute), 136
- NearestNeighbor (*psd\_tools.terminology.Enum* attribute), 136
- Negative (*psd\_tools.terminology.Key* attribute), 165
- NeonGlow (*psd\_tools.terminology.Event* attribute), 148
- NESTED\_SECTION\_DIVIDER\_SETTING (*psd\_tools.constants.Tag* attribute), 82
- NetscapeGray (*psd\_tools.terminology.Enum* attribute), 136
- Neutrals (*psd\_tools.terminology.Enum* attribute), 136
- New (*psd\_tools.terminology.Key* attribute), 165
- new () (*psd\_tools.psd.image\_data.ImageData* class method), 99
- new () (*psd\_tools.psd.image\_resources.ImageResources* class method), 100
- new () (*psd\_tools.PSDImage* class method), 14
- NewView (*psd\_tools.terminology.Enum* attribute), 136
- Next (*psd\_tools.terminology.Enum* attribute), 136
- Next (*psd\_tools.terminology.Event* attribute), 148
- Nikon (*psd\_tools.terminology.Enum* attribute), 136
- Nikon105 (*psd\_tools.terminology.Enum* attribute), 136
- No (*psd\_tools.terminology.Enum* attribute), 136
- NoCompositePS (*psd\_tools.terminology.Enum* attribute), 136
- noise (*psd\_tools.api.effects.DropShadow* attribute), 34
- noise (*psd\_tools.api.effects.InnerGlow* attribute), 37



- noise (*psd\_tools.api.effects.InnerShadow attribute*), 35
- noise (*psd\_tools.api.effects.OuterGlow attribute*), 36
- Noise (*psd\_tools.terminology.Key attribute*), 165
- NON\_BASE (*psd\_tools.constants.Clipping attribute*), 75
- NonImageData (*psd\_tools.terminology.Key attribute*), 165
- NonLinear (*psd\_tools.terminology.Key attribute*), 165
- NORMAL (*psd\_tools.constants.BlendMode attribute*), 74
- Normal (*psd\_tools.terminology.Enum attribute*), 136
- NormalPath (*psd\_tools.terminology.Enum attribute*), 136
- not\_link (*psd\_tools.psd.vector.VectorMaskSetting attribute*), 120
- not\_linked (*psd\_tools.api.shape.VectorMask attribute*), 68
- NotePaper (*psd\_tools.terminology.Event attribute*), 148
- Notify (*psd\_tools.terminology.Event attribute*), 148
- Notify (*psd\_tools.terminology.Type attribute*), 176
- NTSC (*psd\_tools.terminology.Enum attribute*), 136
- NTSCColors (*psd\_tools.terminology.Event attribute*), 148
- Null (*psd\_tools.terminology.Enum attribute*), 136
- Null (*psd\_tools.terminology.Event attribute*), 148
- Null (*psd\_tools.terminology.Key attribute*), 165
- Null (*psd\_tools.terminology.Klass attribute*), 124
- number (*psd\_tools.psd.image\_resources.URLItem attribute*), 106
- Number (*psd\_tools.terminology.Key attribute*), 165
- NumberOfCacheLevels (*psd\_tools.terminology.Key attribute*), 165
- NumberOfCacheLevels64 (*psd\_tools.terminology.Key attribute*), 165
- NumberOfChannels (*psd\_tools.terminology.Key attribute*), 165
- NumberOfChildren (*psd\_tools.terminology.Key attribute*), 165
- NumberOfDocuments (*psd\_tools.terminology.Key attribute*), 165
- NumberOfGenerators (*psd\_tools.terminology.Key attribute*), 165
- NumberOfLayers (*psd\_tools.terminology.Key attribute*), 165
- NumberOfLevels (*psd\_tools.terminology.Key attribute*), 165
- NumberOfPaths (*psd\_tools.terminology.Key attribute*), 165
- NumberOfRipples (*psd\_tools.terminology.Key attribute*), 165
- NumberOfSiblings (*psd\_tools.terminology.Key attribute*), 165
- NumericElement (*class in psd\_tools.psd.base*), 85
- NumLights (*psd\_tools.terminology.Key attribute*), 165
- numpy () (*psd\_tools.api.adjustments.GradientFill method*), 27
- numpy () (*psd\_tools.api.adjustments.PatternFill method*), 23
- numpy () (*psd\_tools.api.adjustments.SolidColorFill method*), 19
- numpy () (*psd\_tools.api.layers.Artboard method*), 44
- numpy () (*psd\_tools.api.layers.Group method*), 48
- numpy () (*psd\_tools.api.layers.PixelLayer method*), 52
- numpy () (*psd\_tools.api.layers.ShapeLayer method*), 56
- numpy () (*psd\_tools.api.layers.SmartObjectLayer method*), 60
- numpy () (*psd\_tools.api.layers.TypeLayer method*), 64
- numpy () (*psd\_tools.PSDImage method*), 14
- ## O
- Object (*psd\_tools.terminology.Type attribute*), 176
- OBJECT\_BASED\_EFFECTS\_LAYER\_INFO (*psd\_tools.constants.Tag attribute*), 83
- OBJECT\_BASED\_EFFECTS\_LAYER\_INFO\_V0 (*psd\_tools.constants.Tag attribute*), 83
- OBJECT\_BASED\_EFFECTS\_LAYER\_INFO\_V1 (*psd\_tools.constants.Tag attribute*), 83
- ObjectArray (*class in psd\_tools.psd.descriptor*), 90
- ObjectName (*psd\_tools.terminology.Key attribute*), 165
- ObjectReference (*psd\_tools.terminology.Type attribute*), 176
- OBSOLETE1 (*psd\_tools.constants.Resource attribute*), 79
- OBSOLETE2 (*psd\_tools.constants.Resource attribute*), 79
- OBSOLETE3 (*psd\_tools.constants.Resource attribute*), 79
- OBSOLETE4 (*psd\_tools.constants.Resource attribute*), 79
- OBSOLETE5 (*psd\_tools.constants.Resource attribute*), 79
- ObsoleteTextLayer (*psd\_tools.terminology.Klass attribute*), 124
- OceanRipple (*psd\_tools.terminology.Event attribute*), 148
- Off (*psd\_tools.terminology.Enum attribute*), 136
- Offset (*class in psd\_tools.psd.descriptor*), 91
- offset (*psd\_tools.api.adjustments.Exposure attribute*), 30
- offset (*psd\_tools.api.adjustments.GradientFill attribute*), 27
- offset (*psd\_tools.api.adjustments.PatternFill attribute*), 23
- offset (*psd\_tools.api.adjustments.SolidColorFill attribute*), 19
- offset (*psd\_tools.api.effects.GradientOverlay attribute*), 38
- offset (*psd\_tools.api.layers.Artboard attribute*), 44

- offset (*psd\_tools.api.layers.Group* attribute), 48
- offset (*psd\_tools.api.layers.PixelLayer* attribute), 52
- offset (*psd\_tools.api.layers.ShapeLayer* attribute), 56
- offset (*psd\_tools.api.layers.SmartObjectLayer* attribute), 60
- offset (*psd\_tools.api.layers.TypeLayer* attribute), 64
- offset (*psd\_tools.PSDImage* attribute), 14
- Offset (*psd\_tools.terminology.Event* attribute), 148
- Offset (*psd\_tools.terminology.Form* attribute), 151
- Offset (*psd\_tools.terminology.Key* attribute), 165
- Offset (*psd\_tools.terminology.Klass* attribute), 124
- OldSmallFontType (*psd\_tools.terminology.Key* attribute), 165
- On (*psd\_tools.terminology.Enum* attribute), 136
- On (*psd\_tools.terminology.Key* attribute), 165
- ONION\_SKINS (*psd\_tools.constants.Resource* attribute), 79
- OnOff (*psd\_tools.terminology.Type* attribute), 176
- opacity (*psd\_tools.api.adjustments.GradientFill* attribute), 27
- opacity (*psd\_tools.api.adjustments.PatternFill* attribute), 23
- opacity (*psd\_tools.api.adjustments.SolidColorFill* attribute), 19
- opacity (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- opacity (*psd\_tools.api.effects.ColorOverlay* attribute), 37
- opacity (*psd\_tools.api.effects.DropShadow* attribute), 34
- opacity (*psd\_tools.api.effects.GradientOverlay* attribute), 38
- opacity (*psd\_tools.api.effects.InnerGlow* attribute), 37
- opacity (*psd\_tools.api.effects.InnerShadow* attribute), 35
- opacity (*psd\_tools.api.effects.OuterGlow* attribute), 36
- opacity (*psd\_tools.api.effects.PatternOverlay* attribute), 38
- opacity (*psd\_tools.api.effects.Satin* attribute), 41
- opacity (*psd\_tools.api.effects.Stroke* attribute), 39
- opacity (*psd\_tools.api.layers.Artboard* attribute), 44
- opacity (*psd\_tools.api.layers.Group* attribute), 48
- opacity (*psd\_tools.api.layers.PixelLayer* attribute), 52
- opacity (*psd\_tools.api.layers.ShapeLayer* attribute), 56
- opacity (*psd\_tools.api.layers.SmartObjectLayer* attribute), 60
- opacity (*psd\_tools.api.layers.TypeLayer* attribute), 64
- opacity (*psd\_tools.api.shape.Stroke* attribute), 69
- opacity (*psd\_tools.psd.effects\_layer.InnerGlowInfo* attribute), 95
- opacity (*psd\_tools.psd.effects\_layer.OuterGlowInfo* attribute), 95
- opacity (*psd\_tools.psd.effects\_layer.ShadowInfo* attribute), 94
- opacity (*psd\_tools.psd.effects\_layer.SolidFillInfo* attribute), 96
- opacity (*psd\_tools.psd.layer\_and\_mask.GlobalLayerMaskInfo* attribute), 107
- opacity (*psd\_tools.psd.layer\_and\_mask.LayerRecord* attribute), 108
- opacity (*psd\_tools.psd.tagged\_blocks.FilterMask* attribute), 116
- opacity (*psd\_tools.psd.tagged\_blocks.UserMask* attribute), 118
- Opacity (*psd\_tools.terminology.Key* attribute), 165
- Opacity (*psd\_tools.terminology.Klass* attribute), 124
- Open (*psd\_tools.terminology.Event* attribute), 148
- open () (*psd\_tools.api.smart\_object.SmartObject* method), 73
- open () (*psd\_tools.PSDImage* class method), 14
- open\_file (*psd\_tools.psd.linked\_layer.LinkedImage* attribute), 113
- OPEN\_FOLDER (*psd\_tools.constants.SectionDivider* attribute), 81
- OPEN\_KNOT\_LINKED (*psd\_tools.constants.PathResourceID* attribute), 76
- OPEN\_KNOT\_UNLINKED (*psd\_tools.constants.PathResourceID* attribute), 77
- OPEN\_LENGTH (*psd\_tools.constants.PathResourceID* attribute), 77
- OpenAs (*psd\_tools.terminology.Enum* attribute), 136
- OpenUntitled (*psd\_tools.terminology.Event* attribute), 148
- operation (*psd\_tools.psd.vector.Subpath* attribute), 119
- Optimized (*psd\_tools.terminology.Key* attribute), 165
- Orange (*psd\_tools.terminology.Enum* attribute), 136
- Ordinal (*psd\_tools.terminology.Type* attribute), 176
- Orientation (*psd\_tools.terminology.Key* attribute), 165
- Orientation (*psd\_tools.terminology.Type* attribute), 176
- origin (*psd\_tools.psd.image\_resources.SliceV6* attribute), 104
- ORIGIN\_PATH\_INFO (*psd\_tools.constants.Resource* attribute), 79
- origin\_type (*psd\_tools.api.shape.Ellipse* attribute), 71
- origin\_type (*psd\_tools.api.shape.Line* attribute), 70
- origin\_type (*psd\_tools.api.shape.Rectangle* attribute), 71
- origin\_type (*psd\_tools.api.shape.RoundedRectangle* attribute), 72
- OriginalHeader (*psd\_tools.terminology.Key* attribute), 165

- OriginalTransmissionReference  
(*psd\_tools.terminology.Key* attribute), 165
- origination (*psd\_tools.api.adjustments.GradientFill* attribute), 28
- origination (*psd\_tools.api.adjustments.PatternFill* attribute), 23
- origination (*psd\_tools.api.adjustments.SolidColorFill* attribute), 20
- origination (*psd\_tools.api.layers.Artboard* attribute), 44
- origination (*psd\_tools.api.layers.Group* attribute), 48
- origination (*psd\_tools.api.layers.PixelLayer* attribute), 52
- origination (*psd\_tools.api.layers.ShapeLayer* attribute), 56
- origination (*psd\_tools.api.layers.SmartObjectLayer* attribute), 60
- origination (*psd\_tools.api.layers.TypeLayer* attribute), 64
- OS2 (*psd\_tools.terminology.Enum* attribute), 136
- OTHER (*psd\_tools.constants.SectionDivider* attribute), 81
- OtherCursors (*psd\_tools.terminology.Key* attribute), 166
- OUTER\_GLOW (*psd\_tools.constants.EffectOSType* attribute), 76
- OuterBevel (*psd\_tools.terminology.Enum* attribute), 136
- OuterGlow (*class in psd\_tools.api.effects*), 35
- OuterGlow (*psd\_tools.terminology.Key* attribute), 166
- OuterGlow (*psd\_tools.terminology.Klass* attribute), 124
- OuterGlowInfo (*class in psd\_tools.psd.effects\_layer*), 94
- OutFromCenter (*psd\_tools.terminology.Enum* attribute), 136
- OutOfGamut (*psd\_tools.terminology.Enum* attribute), 136
- Output (*psd\_tools.terminology.Key* attribute), 166
- OutputBlackPoint (*psd\_tools.terminology.Key* attribute), 166
- OutputWhitePoint (*psd\_tools.terminology.Key* attribute), 166
- OutsetFrame (*psd\_tools.terminology.Enum* attribute), 136
- Outside (*psd\_tools.terminology.Enum* attribute), 136
- OVERLAY (*psd\_tools.constants.BlendMode* attribute), 74
- Overlay (*psd\_tools.terminology.Enum* attribute), 136
- overlay\_color (*psd\_tools.psd.layer\_and\_mask.GlobalLayerMask* attribute), 107
- overprint (*psd\_tools.api.effects.Stroke* attribute), 39
- OverprintColors (*psd\_tools.terminology.Key* attribute), 166
- OverrideOpen (*psd\_tools.terminology.Key* attribute), 166
- OverridePrinter (*psd\_tools.terminology.Key* attribute), 166
- OverrideSave (*psd\_tools.terminology.Key* attribute), 166
- ## P
- P22EBU (*psd\_tools.terminology.Enum* attribute), 136
- PageFormat (*psd\_tools.terminology.Key* attribute), 166
- PageNumber (*psd\_tools.terminology.Key* attribute), 166
- PagePosCentered (*psd\_tools.terminology.Enum* attribute), 137
- PagePosition (*psd\_tools.terminology.Key* attribute), 166
- PagePosition (*psd\_tools.terminology.Type* attribute), 176
- PagePosTopLeft (*psd\_tools.terminology.Enum* attribute), 137
- PageSetup (*psd\_tools.terminology.Enum* attribute), 137
- PageSetup (*psd\_tools.terminology.Key* attribute), 166
- PageSetup (*psd\_tools.terminology.Klass* attribute), 124
- PaintbrushEraser (*psd\_tools.terminology.Enum* attribute), 137
- PaintbrushTool (*psd\_tools.terminology.Klass* attribute), 124
- PaintCursorKind (*psd\_tools.terminology.Key* attribute), 166
- PaintDaubs (*psd\_tools.terminology.Event* attribute), 148
- PaintingCursors (*psd\_tools.terminology.Key* attribute), 166
- PaintType (*psd\_tools.terminology.Key* attribute), 166
- Palette (*psd\_tools.terminology.Key* attribute), 166
- PaletteFile (*psd\_tools.terminology.Key* attribute), 166
- PaletteKnife (*psd\_tools.terminology.Event* attribute), 148
- PalSecam (*psd\_tools.terminology.Enum* attribute), 137
- PanaVision (*psd\_tools.terminology.Enum* attribute), 137
- PaperBrightness (*psd\_tools.terminology.Key* attribute), 166
- parameters (*psd\_tools.api.mask.Mask* attribute), 67
- parameters (*psd\_tools.psd.layer\_and\_mask.MaskData* attribute), 110
- parameters\_applied  
(*psd\_tools.psd.layer\_and\_mask.MaskFlags* attribute), 110

- parent (*psd\_tools.api.adjustments.GradientFill attribute*), 28
- parent (*psd\_tools.api.adjustments.PatternFill attribute*), 24
- parent (*psd\_tools.api.adjustments.SolidColorFill attribute*), 20
- parent (*psd\_tools.api.layers.Artboard attribute*), 44
- parent (*psd\_tools.api.layers.Group attribute*), 48
- parent (*psd\_tools.api.layers.PixelLayer attribute*), 52
- parent (*psd\_tools.api.layers.ShapeLayer attribute*), 56
- parent (*psd\_tools.api.layers.SmartObjectLayer attribute*), 60
- parent (*psd\_tools.api.layers.TypeLayer attribute*), 65
- parent (*psd\_tools.PSDImage attribute*), 15
- ParentIndex (*psd\_tools.terminology.Key attribute*), 166
- ParentName (*psd\_tools.terminology.Key attribute*), 166
- PascalString (class in *psd\_tools.psd.image\_resources*), 102
- PASS\_THROUGH (*psd\_tools.constants.BlendMode attribute*), 74
- Paste (*psd\_tools.terminology.Event attribute*), 148
- PasteEffects (*psd\_tools.terminology.Event attribute*), 148
- PasteInto (*psd\_tools.terminology.Event attribute*), 148
- PasteOutside (*psd\_tools.terminology.Event attribute*), 148
- Patchwork (*psd\_tools.terminology.Event attribute*), 148
- Path (class in *psd\_tools.psd.descriptor*), 91
- Path (class in *psd\_tools.psd.vector*), 118
- path (*psd\_tools.psd.vector.VectorMaskSetting attribute*), 120
- Path (*psd\_tools.terminology.Key attribute*), 166
- Path (*psd\_tools.terminology.Klass attribute*), 124
- PATH\_FILL (*psd\_tools.constants.PathResourceID attribute*), 77
- PATH\_INFO\_0 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_1 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_2 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_3 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_4 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_5 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_6 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_7 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_8 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_9 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_990 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_991 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_992 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_993 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_994 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_995 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_996 (*psd\_tools.constants.Resource attribute*), 79
- PATH\_INFO\_997 (*psd\_tools.constants.Resource attribute*), 80
- PATH\_SELECTION\_STATE (*psd\_tools.constants.Resource attribute*), 80
- PathComponent (*psd\_tools.terminology.Klass attribute*), 124
- PathContents (*psd\_tools.terminology.Key attribute*), 166
- PathFillRule (class in *psd\_tools.psd.vector*), 120
- PathKind (*psd\_tools.terminology.Type attribute*), 176
- PathName (*psd\_tools.terminology.Key attribute*), 166
- PathPoint (*psd\_tools.terminology.Klass attribute*), 124
- PathReference (*psd\_tools.terminology.Type attribute*), 176
- PathResourceID (class in *psd\_tools.constants*), 76
- paths (*psd\_tools.api.shape.VectorMask attribute*), 68
- PathsPaletteOptions (*psd\_tools.terminology.Enum attribute*), 137
- PATT (*psd\_tools.constants.Tag attribute*), 83
- Pattern (class in *psd\_tools.psd.patterns*), 113
- pattern (*psd\_tools.api.effects.PatternOverlay attribute*), 38
- pattern (*psd\_tools.api.effects.Stroke attribute*), 39
- Pattern (*psd\_tools.terminology.Enum attribute*), 137
- Pattern (*psd\_tools.terminology.Key attribute*), 166
- Pattern (*psd\_tools.terminology.Klass attribute*), 124
- PATTERN\_DATA (*psd\_tools.constants.Tag attribute*), 83
- PATTERN\_FILL\_SETTING (*psd\_tools.constants.Tag attribute*), 83
- pattern\_id (*psd\_tools.psd.patterns.Pattern attribute*), 113
- PatternDither (*psd\_tools.terminology.Enum attribute*), 79



- tribute*), 137
- PatternFill (*class in psd\_tools.api.adjustments*), 21
- PatternOverlay (*class in psd\_tools.api.effects*), 38
- Patterns (*class in psd\_tools.psd.patterns*), 113
- PATTERNS1 (*psd\_tools.constants.Tag attribute*), 83
- PATTERNS2 (*psd\_tools.constants.Tag attribute*), 83
- PATTERNS3 (*psd\_tools.constants.Tag attribute*), 83
- PatternStampTool (*psd\_tools.terminology.Klass attribute*), 124
- PDFGenericFormat (*psd\_tools.terminology.Klass attribute*), 124
- PencilEraser (*psd\_tools.terminology.Enum attribute*), 137
- PencilTool (*psd\_tools.terminology.Klass attribute*), 124
- PencilWidth (*psd\_tools.terminology.Key attribute*), 166
- PER\_LAYER (*psd\_tools.constants.GlobalLayerMaskKind attribute*), 76
- Percent (*psd\_tools.terminology.Unit attribute*), 178
- Perceptual (*psd\_tools.terminology.Enum attribute*), 137
- Perspective (*psd\_tools.terminology.Enum attribute*), 137
- PerspectiveIndex (*psd\_tools.terminology.Key attribute*), 166
- phase (*psd\_tools.api.effects.PatternOverlay attribute*), 38
- Phosphors (*psd\_tools.terminology.Key attribute*), 166
- Phosphors (*psd\_tools.terminology.Type attribute*), 176
- PhosphorsCustomPhosphors (*psd\_tools.terminology.Type attribute*), 176
- PHOTO\_FILTER (*psd\_tools.constants.Tag attribute*), 83
- Photocopy (*psd\_tools.terminology.Event attribute*), 148
- PhotoFilter (*class in psd\_tools.api.adjustments*), 32
- Photoshop20Format (*psd\_tools.terminology.Klass attribute*), 125
- Photoshop35Format (*psd\_tools.terminology.Klass attribute*), 125
- PhotoshopDCS2Format (*psd\_tools.terminology.Klass attribute*), 125
- PhotoshopDCSFormat (*psd\_tools.terminology.Klass attribute*), 125
- PhotoshopEPSFormat (*psd\_tools.terminology.Klass attribute*), 125
- PhotoshopPDFFormat (*psd\_tools.terminology.Klass attribute*), 125
- PhotoshopPicker (*psd\_tools.terminology.Enum attribute*), 137
- PickCMYK (*psd\_tools.terminology.Enum attribute*), 137
- PickerID (*psd\_tools.terminology.Key attribute*), 166
- PickerKind (*psd\_tools.terminology.Key attribute*), 166
- PickerKind (*psd\_tools.terminology.Type attribute*), 176
- PickGray (*psd\_tools.terminology.Enum attribute*), 137
- PickHSB (*psd\_tools.terminology.Enum attribute*), 137
- PickLab (*psd\_tools.terminology.Enum attribute*), 137
- PickOptions (*psd\_tools.terminology.Enum attribute*), 137
- PickRGB (*psd\_tools.terminology.Enum attribute*), 137
- PICTFileFormat (*psd\_tools.terminology.Klass attribute*), 124
- PICTResourceFormat (*psd\_tools.terminology.Klass attribute*), 124
- PillowEmboss (*psd\_tools.terminology.Enum attribute*), 137
- PIN\_LIGHT (*psd\_tools.constants.BlendMode attribute*), 74
- Pinch (*psd\_tools.terminology.Event attribute*), 148
- Pixel (*psd\_tools.terminology.Klass attribute*), 125
- PIXEL\_ASPECT\_RATIO (*psd\_tools.constants.Resource attribute*), 80
- pixel\_data\_irrelevant (*psd\_tools.psd.layer\_and\_mask.LayerFlags attribute*), 109
- pixel\_depth (*psd\_tools.psd.patterns.VirtualMemoryArray attribute*), 114
- PIXEL\_SOURCE\_DATA1 (*psd\_tools.constants.Tag attribute*), 83
- PIXEL\_SOURCE\_DATA2 (*psd\_tools.constants.Tag attribute*), 83
- PixelAspectRatio (*class in psd\_tools.psd.image\_resources*), 102
- PixelLayer (*class in psd\_tools.api.layers*), 50
- PixelPaintFormat (*psd\_tools.terminology.Klass attribute*), 125
- PixelPaintSize (*psd\_tools.terminology.Key attribute*), 166
- PixelPaintSize (*psd\_tools.terminology.Type attribute*), 176
- PixelPaintSize1 (*psd\_tools.terminology.Enum attribute*), 137
- PixelPaintSize2 (*psd\_tools.terminology.Enum attribute*), 137
- PixelPaintSize3 (*psd\_tools.terminology.Enum attribute*), 137
- PixelPaintSize4 (*psd\_tools.terminology.Enum attribute*), 137
- Pixels (*psd\_tools.terminology.Unit attribute*), 178
- PixelSourceData2 (*class in psd\_tools.psd.tagged\_blocks*), 116
- Place (*psd\_tools.terminology.Enum attribute*), 137
- Place (*psd\_tools.terminology.Event attribute*), 148
- PLACED\_LAYER1 (*psd\_tools.constants.Tag attribute*), 83

- PLACED\_LAYER2 (*psd\_tools.constants.Tag* attribute), 83
- PlacedLayerData (class in *psd\_tools.psd.tagged\_blocks*), 116
- PlacedLayerType (class in *psd\_tools.constants*), 77
- planes (*psd\_tools.psd.image\_resources.ThumbnailResource* attribute), 105
- Plaster (*psd\_tools.terminology.Event* attribute), 148
- PlasticWrap (*psd\_tools.terminology.Event* attribute), 148
- Platform (*psd\_tools.terminology.Key* attribute), 166
- Platform (*psd\_tools.terminology.Type* attribute), 177
- Play (*psd\_tools.terminology.Event* attribute), 148
- PlaybackOptions (*psd\_tools.terminology.Enum* attribute), 137
- PLUGIN\_RESOURCE\_0 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_1 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_2 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_3 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_4 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_4990 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_4991 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_4992 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_4993 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_4994 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_4995 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_4996 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_4997 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_4998 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_4999 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_5 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_6 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_7 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_8 (*psd\_tools.constants.Resource* attribute), 80
- PLUGIN\_RESOURCE\_9 (*psd\_tools.constants.Resource* attribute), 80
- PluginFolder (*psd\_tools.terminology.Key* attribute), 166
- PluginPicker (*psd\_tools.terminology.Enum* attribute), 137
- PluginPrefs (*psd\_tools.terminology.Key* attribute), 166
- PluginPrefs (*psd\_tools.terminology.Klass* attribute), 125
- PluginsScratchDiskPreferences (*psd\_tools.terminology.Enum* attribute), 137
- PNGFilter (*psd\_tools.terminology.Key* attribute), 166
- PNGFilter (*psd\_tools.terminology.Type* attribute), 176
- PNGFilterAdaptive (*psd\_tools.terminology.Enum* attribute), 136
- PNGFilterAverage (*psd\_tools.terminology.Enum* attribute), 136
- PNGFilterNone (*psd\_tools.terminology.Enum* attribute), 137
- PNGFilterPaeth (*psd\_tools.terminology.Enum* attribute), 137
- PNGFilterSub (*psd\_tools.terminology.Enum* attribute), 137
- PNGFilterUp (*psd\_tools.terminology.Enum* attribute), 137
- PNGFormat (*psd\_tools.terminology.Klass* attribute), 124
- PNGInterlaceAdam7 (*psd\_tools.terminology.Enum* attribute), 137
- PNGInterlaceNone (*psd\_tools.terminology.Enum* attribute), 137
- PNGInterlaceType (*psd\_tools.terminology.Key* attribute), 166
- PNGInterlaceType (*psd\_tools.terminology.Type* attribute), 176
- point (*psd\_tools.psd.patterns.Pattern* attribute), 113
- Point (*psd\_tools.terminology.Klass* attribute), 125
- Point16 (*psd\_tools.terminology.Klass* attribute), 125
- Pointillize (*psd\_tools.terminology.Event* attribute), 148
- Points (*psd\_tools.terminology.Key* attribute), 167
- Points (*psd\_tools.terminology.Unit* attribute), 178
- Polar (*psd\_tools.terminology.Event* attribute), 148

- PolarToRect (*psd\_tools.terminology.Enum* attribute), 137
- Polygon (*psd\_tools.terminology.Klass* attribute), 125
- PondRipples (*psd\_tools.terminology.Enum* attribute), 137
- pos\_relative\_to\_layer (*psd\_tools.psd.layer\_and\_mask.MaskFlags* attribute), 110
- position (*psd\_tools.api.effects.Stroke* attribute), 39
- Position (*psd\_tools.terminology.Key* attribute), 167
- PosterEdges (*psd\_tools.terminology.Event* attribute), 148
- Posterization (*psd\_tools.terminology.Key* attribute), 167
- Posterize (*class* in *psd\_tools.api.adjustments*), 32
- posterize (*psd\_tools.api.adjustments.Posterize* attribute), 32
- POSTERIZE (*psd\_tools.constants.Tag* attribute), 83
- Posterize (*psd\_tools.terminology.Event* attribute), 148
- Posterize (*psd\_tools.terminology.Klass* attribute), 125
- PostScriptColor (*psd\_tools.terminology.Key* attribute), 167
- preceding (*psd\_tools.psd.vector.Knot* attribute), 119
- Precise (*psd\_tools.terminology.Enum* attribute), 138
- PreciseMatte (*psd\_tools.terminology.Enum* attribute), 138
- PredefinedColors (*psd\_tools.terminology.Key* attribute), 167
- PreferBuiltin (*psd\_tools.terminology.Key* attribute), 167
- Preferences (*psd\_tools.terminology.Key* attribute), 167
- Preferences (*psd\_tools.terminology.Klass* attribute), 125
- present (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- present (*psd\_tools.api.effects.ColorOverlay* attribute), 37
- present (*psd\_tools.api.effects.DropShadow* attribute), 34
- present (*psd\_tools.api.effects.GradientOverlay* attribute), 38
- present (*psd\_tools.api.effects.InnerGlow* attribute), 37
- present (*psd\_tools.api.effects.InnerShadow* attribute), 35
- present (*psd\_tools.api.effects.OuterGlow* attribute), 36
- present (*psd\_tools.api.effects.PatternOverlay* attribute), 38
- present (*psd\_tools.api.effects.Satin* attribute), 41
- present (*psd\_tools.api.effects.Stroke* attribute), 39
- PreserveAdditional (*psd\_tools.terminology.Key* attribute), 167
- PreserveLuminosity (*psd\_tools.terminology.Key* attribute), 167
- PreserveTransparency (*psd\_tools.terminology.Key* attribute), 167
- preset\_file\_name (*psd\_tools.api.adjustments.BlackAndWhite* attribute), 32
- preset\_kind (*psd\_tools.api.adjustments.BlackAndWhite* attribute), 32
- Pressure (*psd\_tools.terminology.Key* attribute), 167
- Preview (*psd\_tools.terminology.Key* attribute), 167
- Preview (*psd\_tools.terminology.Type* attribute), 177
- PreviewBlack (*psd\_tools.terminology.Enum* attribute), 138
- PreviewCMY (*psd\_tools.terminology.Enum* attribute), 138
- PreviewCMYK (*psd\_tools.terminology.Enum* attribute), 138
- PreviewCMYK (*psd\_tools.terminology.Key* attribute), 167
- PreviewCMYK (*psd\_tools.terminology.Type* attribute), 177
- PreviewCyan (*psd\_tools.terminology.Enum* attribute), 138
- PreviewFullSize (*psd\_tools.terminology.Key* attribute), 167
- PreviewIcon (*psd\_tools.terminology.Key* attribute), 167
- PreviewMacThumbnail (*psd\_tools.terminology.Key* attribute), 167
- PreviewMagenta (*psd\_tools.terminology.Enum* attribute), 138
- PreviewOff (*psd\_tools.terminology.Enum* attribute), 138
- PreviewsQuery (*psd\_tools.terminology.Key* attribute), 167
- PreviewWinThumbnail (*psd\_tools.terminology.Key* attribute), 167
- PreviewYellow (*psd\_tools.terminology.Enum* attribute), 138
- Previous (*psd\_tools.terminology.Enum* attribute), 138
- Previous (*psd\_tools.terminology.Event* attribute), 148
- Primaries (*psd\_tools.terminology.Enum* attribute), 138
- Print (*psd\_tools.terminology.Event* attribute), 148
- PRINT\_FLAGS (*psd\_tools.constants.Resource* attribute), 80
- PRINT\_FLAGS\_INFO (*psd\_tools.constants.Resource* attribute), 80
- PRINT\_INFO\_CS2 (*psd\_tools.constants.Resource* attribute), 80
- PRINT\_INFO\_CS5 (*psd\_tools.constants.Resource* attribute), 80
- PRINT\_SCALE (*psd\_tools.constants.Resource* at-

- tribute*), 80
  - PRINT\_STYLE (*psd\_tools.constants.Resource attribute*), 80
  - PrintFlags (*class in psd\_tools.psd.image\_resources*), 103
  - PrintFlagsInfo (*class in psd\_tools.psd.image\_resources*), 103
  - PrintingInksSetup (*psd\_tools.terminology.Enum attribute*), 138
  - PrintScale (*class in psd\_tools.psd.image\_resources*), 103
  - PrintScaleStyle (*class in psd\_tools.constants*), 77
  - PrintSettings (*psd\_tools.terminology.Key attribute*), 167
  - PrintSize (*psd\_tools.terminology.Enum attribute*), 138
  - ProfileMismatch (*psd\_tools.terminology.Type attribute*), 177
  - ProfileSetup (*psd\_tools.terminology.Key attribute*), 167
  - ProfileSetup (*psd\_tools.terminology.Klass attribute*), 125
  - ProfileToProfile (*psd\_tools.terminology.Event attribute*), 149
  - Property (*class in psd\_tools.psd.descriptor*), 90
  - Property (*class in psd\_tools.psd.engine\_data*), 93
  - Property (*psd\_tools.terminology.Form attribute*), 151
  - Property (*psd\_tools.terminology.Klass attribute*), 125
  - PROTECTED\_SETTING (*psd\_tools.constants.Tag attribute*), 83
  - ProtectedSetting (*class in psd\_tools.psd.tagged\_blocks*), 116
  - ProvinceState (*psd\_tools.terminology.Key attribute*), 167
  - PSD (*class in psd\_tools.psd*), 84
  - psd\_tools.api.adjustments (*module*), 17
  - psd\_tools.api.effects (*module*), 34
  - psd\_tools.api.layers (*module*), 41
  - psd\_tools.api.mask (*module*), 66
  - psd\_tools.api.shape (*module*), 67
  - psd\_tools.api.smart\_object (*module*), 72
  - psd\_tools.constants (*module*), 73
  - psd\_tools.psd (*module*), 84
  - psd\_tools.psd.base (*module*), 84
  - psd\_tools.psd.color\_mode\_data (*module*), 86
  - psd\_tools.psd.descriptor (*module*), 87
  - psd\_tools.psd.effects\_layer (*module*), 93
  - psd\_tools.psd.engine\_data (*module*), 92
  - psd\_tools.psd.filter\_effects (*module*), 96
  - psd\_tools.psd.header (*module*), 97
  - psd\_tools.psd.image\_data (*module*), 98
  - psd\_tools.psd.image\_resources (*module*), 99
  - psd\_tools.psd.layer\_and\_mask (*module*), 106
  - psd\_tools.psd.linked\_layer (*module*), 112
  - psd\_tools.psd.patterns (*module*), 113
  - psd\_tools.psd.tagged\_blocks (*module*), 114
  - psd\_tools.psd.vector (*module*), 118
  - psd\_tools.terminology (*module*), 120
  - PSDImage (*class in psd\_tools*), 12
  - Purge (*psd\_tools.terminology.Event attribute*), 149
  - PurgeItem (*psd\_tools.terminology.Type attribute*), 177
  - Purple (*psd\_tools.terminology.Enum attribute*), 138
  - Pyramids (*psd\_tools.terminology.Enum attribute*), 138
- ## Q
- QCSAverage (*psd\_tools.terminology.Enum attribute*), 138
  - QCSCorner0 (*psd\_tools.terminology.Enum attribute*), 138
  - QCSCorner1 (*psd\_tools.terminology.Enum attribute*), 138
  - QCSCorner2 (*psd\_tools.terminology.Enum attribute*), 138
  - QCSCorner3 (*psd\_tools.terminology.Enum attribute*), 138
  - QCSIndependent (*psd\_tools.terminology.Enum attribute*), 138
  - QCSSide0 (*psd\_tools.terminology.Enum attribute*), 138
  - QCSSide1 (*psd\_tools.terminology.Enum attribute*), 138
  - QCSSide2 (*psd\_tools.terminology.Enum attribute*), 138
  - QCSSide3 (*psd\_tools.terminology.Enum attribute*), 138
  - QuadCenterState (*psd\_tools.terminology.Type attribute*), 177
  - Quadtone (*psd\_tools.terminology.Enum attribute*), 138
  - Quality (*psd\_tools.terminology.Key attribute*), 167
  - Quality (*psd\_tools.terminology.Type attribute*), 177
  - quality\_jitter (*psd\_tools.api.effects.InnerGlow attribute*), 37
  - quality\_jitter (*psd\_tools.api.effects.OuterGlow attribute*), 36
  - quality\_range (*psd\_tools.api.effects.InnerGlow attribute*), 37
  - quality\_range (*psd\_tools.api.effects.OuterGlow attribute*), 36
  - QueryAlways (*psd\_tools.terminology.Enum attribute*), 138
  - QueryAsk (*psd\_tools.terminology.Enum attribute*), 138
  - QueryNever (*psd\_tools.terminology.Enum attribute*), 138
  - QueryState (*psd\_tools.terminology.Type attribute*), 177
  - QUICK\_MASK\_INFO (*psd\_tools.constants.Resource attribute*), 80
  - QuickMask (*psd\_tools.terminology.Key attribute*), 167
  - Quit (*psd\_tools.terminology.Event attribute*), 149
- ## R
- Radial (*psd\_tools.terminology.Enum attribute*), 138



- RadialBlur (*psd\_tools.terminology.Event attribute*), 149
- radii (*psd\_tools.api.shape.RoundedRectangle attribute*), 72
- Radius (*psd\_tools.terminology.Key attribute*), 167
- Random (*psd\_tools.terminology.Enum attribute*), 138
- random\_seed (*psd\_tools.api.adjustments.GradientMap attribute*), 33
- RandomSeed (*psd\_tools.terminology.Key attribute*), 167
- Range (*psd\_tools.terminology.Klass attribute*), 125
- RASTER (*psd\_tools.constants.PlacedLayerType attribute*), 77
- Rasterize (*psd\_tools.terminology.Event attribute*), 149
- RasterizeTypeSheet (*psd\_tools.terminology.Event attribute*), 149
- Ratio (*psd\_tools.terminology.Key attribute*), 167
- RAW (*psd\_tools.constants.Compression attribute*), 75
- RawData (*class in psd\_tools.psd.descriptor*), 91
- RawData (*psd\_tools.terminology.Type attribute*), 177
- RawFormat (*psd\_tools.terminology.Klass attribute*), 125
- read() (*psd\_tools.psd.base.BaseElement class method*), 85
- reader (*psd\_tools.psd.image\_resources.VersionInfo attribute*), 106
- real\_background\_color (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 110
- real\_bottom (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 110
- real\_flags (*psd\_tools.api.mask.Mask attribute*), 67
- real\_flags (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 110
- real\_height (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 110
- real\_highlight\_color (*psd\_tools.psd.effects\_layer.BevelInfo attribute*), 96
- real\_left (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 110
- real\_right (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 110
- real\_shadow\_color (*psd\_tools.psd.effects\_layer.BevelInfo attribute*), 96
- real\_top (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 110
- REAL\_USER\_LAYER\_MASK (*psd\_tools.constants.ChannelID attribute*), 74
- real\_width (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 110
- RecentFiles (*psd\_tools.terminology.Key attribute*), 167
- Rect16 (*psd\_tools.terminology.Klass attribute*), 125
- Rectangle (*class in psd\_tools.api.shape*), 71
- rectangle (*psd\_tools.psd.filter\_effects.FilterEffect attribute*), 97
- rectangle (*psd\_tools.psd.filter\_effects.FilterEffectExtra attribute*), 97
- rectangle (*psd\_tools.psd.patterns.VirtualMemoryArray attribute*), 114
- rectangle (*psd\_tools.psd.patterns.VirtualMemoryArrayList attribute*), 114
- Rectangle (*psd\_tools.terminology.Klass attribute*), 125
- RectToPolar (*psd\_tools.terminology.Enum attribute*), 138
- red (*psd\_tools.api.adjustments.BlackAndWhite attribute*), 32
- red (*psd\_tools.psd.image\_resources.SliceV6 attribute*), 105
- Red (*psd\_tools.terminology.Enum attribute*), 138
- Red (*psd\_tools.terminology.Key attribute*), 167
- RedBlackPoint (*psd\_tools.terminology.Key attribute*), 167
- RedGamma (*psd\_tools.terminology.Key attribute*), 167
- RedrawComplete (*psd\_tools.terminology.Enum attribute*), 139
- Reds (*psd\_tools.terminology.Enum attribute*), 139
- RedWhitePoint (*psd\_tools.terminology.Key attribute*), 167
- RedX (*psd\_tools.terminology.Key attribute*), 167
- RedY (*psd\_tools.terminology.Key attribute*), 167
- Reference (*class in psd\_tools.psd.descriptor*), 91
- REFERENCE\_POINT (*psd\_tools.constants.Tag attribute*), 83
- ReferencePoint (*class in psd\_tools.psd.tagged\_blocks*), 117
- Reflected (*psd\_tools.terminology.Enum attribute*), 139
- RegistrationMarks (*psd\_tools.terminology.Key attribute*), 167
- Relative (*psd\_tools.terminology.Enum attribute*), 139
- Relative (*psd\_tools.terminology.Key attribute*), 167
- Relief (*psd\_tools.terminology.Key attribute*), 168
- RemoveBlackMatte (*psd\_tools.terminology.Event attribute*), 149
- RemoveLayerMask (*psd\_tools.terminology.Event attribute*), 149
- RemoveWhiteMatte (*psd\_tools.terminology.Event attribute*), 149
- Rename (*psd\_tools.terminology.Event attribute*), 149
- RenderFidelity (*psd\_tools.terminology.Key attribute*), 168
- Repeat (*psd\_tools.terminology.Enum attribute*), 139

- RepeatEdgePixels (*psd\_tools.terminology.Enum attribute*), 139
- ReplaceColor (*psd\_tools.terminology.Event attribute*), 149
- Resample (*psd\_tools.terminology.Key attribute*), 168
- Reset (*psd\_tools.terminology.Event attribute*), 149
- ResizeWindowsOnZoom (*psd\_tools.terminology.Key attribute*), 168
- resolution (*psd\_tools.api.shape.Ellipse attribute*), 71
- resolution (*psd\_tools.api.shape.Line attribute*), 70
- resolution (*psd\_tools.api.shape.Rectangle attribute*), 71
- resolution (*psd\_tools.api.shape.RoundedRectangle attribute*), 72
- resolution (*psd\_tools.api.smart\_object.SmartObject attribute*), 73
- resolution (*psd\_tools.psd.vector.ClipboardRecord attribute*), 119
- Resolution (*psd\_tools.terminology.Key attribute*), 168
- RESOLUTION\_INFO (*psd\_tools.constants.Resource attribute*), 80
- ResoulutionInfo (class in *psd\_tools.psd.image\_resources*), 103
- Resource (class in *psd\_tools.constants*), 77
- resource\_dict (*psd\_tools.api.layers.TypeLayer attribute*), 65
- ResourceID (*psd\_tools.terminology.Key attribute*), 168
- Response (*psd\_tools.terminology.Key attribute*), 168
- RetainHeader (*psd\_tools.terminology.Key attribute*), 168
- Reticulation (*psd\_tools.terminology.Event attribute*), 149
- RevealAll (*psd\_tools.terminology.Enum attribute*), 139
- RevealSelection (*psd\_tools.terminology.Enum attribute*), 139
- Reverse (*psd\_tools.terminology.Key attribute*), 168
- reversed (*psd\_tools.api.adjustments.GradientMap attribute*), 33
- reversed (*psd\_tools.api.effects.GradientOverlay attribute*), 38
- Revert (*psd\_tools.terminology.Enum attribute*), 139
- Revert (*psd\_tools.terminology.Event attribute*), 149
- RGB (*psd\_tools.constants.ColorMode attribute*), 75
- RGB (*psd\_tools.constants.ColorSpaceID attribute*), 75
- RGB (*psd\_tools.terminology.Enum attribute*), 138
- RGB48 (*psd\_tools.terminology.Enum attribute*), 138
- RGBColor (*psd\_tools.terminology.Enum attribute*), 138
- RGBColor (*psd\_tools.terminology.Klass attribute*), 125
- RGBColorMode (*psd\_tools.terminology.Klass attribute*), 125
- RGBSetup (*psd\_tools.terminology.Key attribute*), 167
- RGBSetup (*psd\_tools.terminology.Klass attribute*), 125
- RGBSetupSource (*psd\_tools.terminology.Type attribute*), 177
- right (*psd\_tools.api.adjustments.GradientFill attribute*), 28
- right (*psd\_tools.api.adjustments.PatternFill attribute*), 24
- right (*psd\_tools.api.adjustments.SolidColorFill attribute*), 20
- right (*psd\_tools.api.layers.Artboard attribute*), 44
- right (*psd\_tools.api.layers.PixelLayer attribute*), 52
- right (*psd\_tools.api.layers.ShapeLayer attribute*), 56
- right (*psd\_tools.api.layers.SmartObjectLayer attribute*), 60
- right (*psd\_tools.api.layers.TypeLayer attribute*), 65
- right (*psd\_tools.api.mask.Mask attribute*), 67
- right (*psd\_tools.psd.layer\_and\_mask.LayerRecord attribute*), 108
- right (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 109
- right (*psd\_tools.psd.tagged\_blocks.TypeToolObjectSetting attribute*), 118
- right (*psd\_tools.psd.vector.ClipboardRecord attribute*), 119
- right (*psd\_tools.PSDImage attribute*), 15
- Right (*psd\_tools.terminology.Enum attribute*), 139
- Right (*psd\_tools.terminology.Key attribute*), 168
- Ripple (*psd\_tools.terminology.Event attribute*), 149
- RippleMagnitude (*psd\_tools.terminology.Key attribute*), 168
- RippleSize (*psd\_tools.terminology.Key attribute*), 168
- RippleSize (*psd\_tools.terminology.Type attribute*), 177
- RLE (*psd\_tools.constants.Compression attribute*), 75
- Rotate (*psd\_tools.terminology.Enum attribute*), 139
- Rotate (*psd\_tools.terminology.Event attribute*), 149
- Rotate (*psd\_tools.terminology.Key attribute*), 168
- RotoscopingPreferences (*psd\_tools.terminology.Enum attribute*), 139
- roughness (*psd\_tools.api.adjustments.GradientMap attribute*), 33
- RoughPastels (*psd\_tools.terminology.Event attribute*), 149
- Round (*psd\_tools.terminology.Enum attribute*), 139
- RoundedRectangle (class in *psd\_tools.api.shape*), 72
- Roundness (*psd\_tools.terminology.Key attribute*), 168
- row (*psd\_tools.psd.image\_resources.ThumbnailResource attribute*), 105
- rule (*psd\_tools.psd.vector.InitialFillRule attribute*), 120
- RulerCm (*psd\_tools.terminology.Enum attribute*), 139
- RulerInches (*psd\_tools.terminology.Enum attribute*), 139

- 139
- RulerOriginH (*psd\_tools.terminology.Key attribute*), 168
- RulerOriginV (*psd\_tools.terminology.Key attribute*), 168
- RulerPercent (*psd\_tools.terminology.Enum attribute*), 139
- RulerPicas (*psd\_tools.terminology.Enum attribute*), 139
- RulerPixels (*psd\_tools.terminology.Enum attribute*), 139
- RulerPoints (*psd\_tools.terminology.Enum attribute*), 139
- RulerUnits (*psd\_tools.terminology.Key attribute*), 168
- RulerUnits (*psd\_tools.terminology.Type attribute*), 177
- ## S
- Sample3x3 (*psd\_tools.terminology.Enum attribute*), 139
- Sample5x5 (*psd\_tools.terminology.Enum attribute*), 139
- SamplePoint (*psd\_tools.terminology.Enum attribute*), 139
- Satin (*class in psd\_tools.api.effects*), 41
- Saturate (*psd\_tools.terminology.Enum attribute*), 139
- saturation (*psd\_tools.api.adjustments.Vibrance attribute*), 30
- SATURATION (*psd\_tools.constants.BlendMode attribute*), 74
- Saturation (*psd\_tools.terminology.Enum attribute*), 139
- Saturation (*psd\_tools.terminology.Key attribute*), 168
- SaturationTool (*psd\_tools.terminology.Klass attribute*), 125
- Save (*psd\_tools.terminology.Event attribute*), 149
- save () (*psd\_tools.api.smart\_object.SmartObject method*), 73
- save () (*psd\_tools.PSDImage method*), 15
- SaveAndClose (*psd\_tools.terminology.Key attribute*), 168
- SaveComposite (*psd\_tools.terminology.Key attribute*), 168
- Saved (*psd\_tools.terminology.Enum attribute*), 139
- SaveForWeb (*psd\_tools.terminology.Enum attribute*), 139
- SavePaletteLocations (*psd\_tools.terminology.Key attribute*), 168
- SavePaths (*psd\_tools.terminology.Key attribute*), 168
- SavePyramids (*psd\_tools.terminology.Key attribute*), 168
- Saving (*psd\_tools.terminology.Key attribute*), 168
- SAVING\_MERGED\_TRANSPARENCY (*psd\_tools.constants.Tag attribute*), 83
- SAVING\_MERGED\_TRANSPARENCY16 (*psd\_tools.constants.Tag attribute*), 83
- SAVING\_MERGED\_TRANSPARENCY32 (*psd\_tools.constants.Tag attribute*), 83
- SavingFilesPreferences (*psd\_tools.terminology.Enum attribute*), 139
- scale (*psd\_tools.api.effects.Effects attribute*), 34
- scale (*psd\_tools.api.effects.GradientOverlay attribute*), 38
- scale (*psd\_tools.api.effects.PatternOverlay attribute*), 39
- scale (*psd\_tools.psd.image\_resources.PrintScale attribute*), 103
- Scale (*psd\_tools.terminology.Enum attribute*), 139
- Scale (*psd\_tools.terminology.Key attribute*), 168
- ScaleHorizontal (*psd\_tools.terminology.Key attribute*), 168
- ScaleVertical (*psd\_tools.terminology.Key attribute*), 168
- Scaling (*psd\_tools.terminology.Key attribute*), 168
- Scans (*psd\_tools.terminology.Key attribute*), 168
- ScitexCTFormat (*psd\_tools.terminology.Klass attribute*), 125
- ScratchDisks (*psd\_tools.terminology.Key attribute*), 168
- SCREEN (*psd\_tools.constants.BlendMode attribute*), 74
- Screen (*psd\_tools.terminology.Enum attribute*), 139
- ScreenCircle (*psd\_tools.terminology.Enum attribute*), 139
- ScreenDot (*psd\_tools.terminology.Enum attribute*), 139
- ScreenFile (*psd\_tools.terminology.Key attribute*), 168
- ScreenLine (*psd\_tools.terminology.Enum attribute*), 139
- ScreenType (*psd\_tools.terminology.Key attribute*), 168
- ScreenType (*psd\_tools.terminology.Type attribute*), 177
- SECTION\_DIVIDER\_SETTING (*psd\_tools.constants.Tag attribute*), 83
- SectionDivider (*class in psd\_tools.constants*), 81
- SectionDividerSetting (*class in psd\_tools.psd.tagged\_blocks*), 117
- Select (*psd\_tools.terminology.Event attribute*), 149
- SelectedAreas (*psd\_tools.terminology.Enum attribute*), 139
- Selection (*psd\_tools.terminology.Enum attribute*), 139
- Selection (*psd\_tools.terminology.Klass attribute*), 125

- Selective (*psd\_tools.terminology.Enum* attribute), 140
- SELECTIVE\_COLOR (*psd\_tools.constants.Tag* attribute), 83
- SelectiveColor (class in *psd\_tools.api.adjustments*), 33
- SelectiveColor (*psd\_tools.terminology.Event* attribute), 149
- SelectiveColor (*psd\_tools.terminology.Klass* attribute), 125
- Separations (*psd\_tools.terminology.Key* attribute), 168
- SeparationSetup (*psd\_tools.terminology.Enum* attribute), 140
- SeparationTables (*psd\_tools.terminology.Enum* attribute), 140
- SerialString (*psd\_tools.terminology.Key* attribute), 168
- Set (*psd\_tools.terminology.Event* attribute), 149
- set\_data () (*psd\_tools.psd.image\_data.ImageData* method), 99
- set\_data () (*psd\_tools.psd.layer\_and\_mask.ChannelData* method), 112
- set\_data () (*psd\_tools.psd.patterns.VirtualMemoryArray* method), 114
- ShadingIntensity (*psd\_tools.terminology.Key* attribute), 168
- ShadingNoise (*psd\_tools.terminology.Key* attribute), 168
- ShadingShape (*psd\_tools.terminology.Key* attribute), 169
- shadow\_blend\_mode (*psd\_tools.psd.effects\_layer.BevelInfo* attribute), 96
- shadow\_color (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- shadow\_color (*psd\_tools.psd.effects\_layer.BevelInfo* attribute), 96
- shadow\_mode (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- shadow\_opacity (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- shadow\_opacity (*psd\_tools.psd.effects\_layer.BevelInfo* attribute), 96
- ShadowColor (*psd\_tools.terminology.Key* attribute), 169
- ShadowInfo (class in *psd\_tools.psd.effects\_layer*), 94
- ShadowIntensity (*psd\_tools.terminology.Key* attribute), 169
- ShadowLevels (*psd\_tools.terminology.Key* attribute), 169
- ShadowMode (*psd\_tools.terminology.Key* attribute), 169
- ShadowOpacity (*psd\_tools.terminology.Key* attribute), 169
- shadows (*psd\_tools.api.adjustments.ColorBalance* attribute), 31
- Shadows (*psd\_tools.terminology.Enum* attribute), 140
- shape (*psd\_tools.psd.image\_resources.HalftoneScreen* attribute), 102
- Shape (*psd\_tools.terminology.Key* attribute), 169
- Shape (*psd\_tools.terminology.Type* attribute), 177
- ShapeLayer (class in *psd\_tools.api.layers*), 54
- ShapingCurve (*psd\_tools.terminology.Klass* attribute), 125
- Sharpen (*psd\_tools.terminology.Event* attribute), 149
- SharpenEdges (*psd\_tools.terminology.Event* attribute), 149
- SharpenMore (*psd\_tools.terminology.Event* attribute), 149
- SharpenTool (*psd\_tools.terminology.Klass* attribute), 125
- Sharpness (*psd\_tools.terminology.Key* attribute), 169
- Shear (*psd\_tools.terminology.Event* attribute), 149
- ShearEd (*psd\_tools.terminology.Key* attribute), 169
- ShearPoints (*psd\_tools.terminology.Key* attribute), 169
- ShearSt (*psd\_tools.terminology.Key* attribute), 169
- SHEET\_COLOR\_SETTING (*psd\_tools.constants.Tag* attribute), 83
- SHEET\_DISCLOSURE (*psd\_tools.constants.Resource* attribute), 80
- SheetColorSetting (class in *psd\_tools.psd.tagged\_blocks*), 117
- ShiftKey (*psd\_tools.terminology.Key* attribute), 169
- ShiftKeyToolSwitch (*psd\_tools.terminology.Key* attribute), 169
- ShortInteger (class in *psd\_tools.psd.image\_resources*), 102
- ShortIntegerElement (class in *psd\_tools.psd.base*), 86
- ShortLines (*psd\_tools.terminology.Enum* attribute), 140
- ShortNames (*psd\_tools.terminology.Key* attribute), 169
- ShortStrokes (*psd\_tools.terminology.Enum* attribute), 140
- Show (*psd\_tools.terminology.Event* attribute), 149
- show\_transparency (*psd\_tools.api.adjustments.GradientMap* attribute), 33
- ShowEnglishFontNames (*psd\_tools.terminology.Key* attribute), 169
- ShowMenuColors (*psd\_tools.terminology.Key* attribute), 169
- shown (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- shown (*psd\_tools.api.effects.ColorOverlay* attribute), 37
- shown (*psd\_tools.api.effects.DropShadow* attribute), 34



- shown (*psd\_tools.api.effects.GradientOverlay* attribute), 38
- shown (*psd\_tools.api.effects.InnerGlow* attribute), 37
- shown (*psd\_tools.api.effects.InnerShadow* attribute), 35
- shown (*psd\_tools.api.effects.OuterGlow* attribute), 36
- shown (*psd\_tools.api.effects.PatternOverlay* attribute), 39
- shown (*psd\_tools.api.effects.Satin* attribute), 41
- shown (*psd\_tools.api.effects.Stroke* attribute), 39
- ShowToolTips (*psd\_tools.terminology.Key* attribute), 169
- ShowTransparency (*psd\_tools.terminology.Key* attribute), 169
- signature (*psd\_tools.psd.header.FileHeader* attribute), 98
- signature (*psd\_tools.psd.image\_resources.ImageResource* attribute), 100
- signature (*psd\_tools.psd.layer\_and\_mask.LayerRecord* attribute), 108
- Similar (*psd\_tools.terminology.Event* attribute), 149
- Single72Color (*psd\_tools.terminology.Enum* attribute), 140
- Single72Gray (*psd\_tools.terminology.Enum* attribute), 140
- SingleColumn (*psd\_tools.terminology.Klass* attribute), 125
- SingleNoCompositePS (*psd\_tools.terminology.Enum* attribute), 140
- SingleRow (*psd\_tools.terminology.Klass* attribute), 125
- size (*psd\_tools.api.adjustments.GradientFill* attribute), 28
- size (*psd\_tools.api.adjustments.PatternFill* attribute), 24
- size (*psd\_tools.api.adjustments.SolidColorFill* attribute), 20
- size (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- size (*psd\_tools.api.effects.DropShadow* attribute), 34
- size (*psd\_tools.api.effects.InnerGlow* attribute), 37
- size (*psd\_tools.api.effects.InnerShadow* attribute), 35
- size (*psd\_tools.api.effects.OuterGlow* attribute), 36
- size (*psd\_tools.api.effects.Satin* attribute), 41
- size (*psd\_tools.api.effects.Stroke* attribute), 39
- size (*psd\_tools.api.layers.Artboard* attribute), 45
- size (*psd\_tools.api.layers.Group* attribute), 49
- size (*psd\_tools.api.layers.PixelLayer* attribute), 52
- size (*psd\_tools.api.layers.ShapeLayer* attribute), 56
- size (*psd\_tools.api.layers.SmartObjectLayer* attribute), 60
- size (*psd\_tools.api.layers.TypeLayer* attribute), 65
- size (*psd\_tools.api.mask.Mask* attribute), 67
- size (*psd\_tools.psd.image\_resources.ThumbnailResource* attribute), 105
- size (*psd\_tools.PSDImage* attribute), 15
- SIZE\_TO\_FIT (*psd\_tools.constants.PrintScaleStyle* attribute), 77
- SizeKey (*psd\_tools.terminology.Key* attribute), 169
- Skew (*psd\_tools.terminology.Enum* attribute), 140
- Skew (*psd\_tools.terminology.Key* attribute), 169
- slice\_id (*psd\_tools.psd.image\_resources.SliceV6* attribute), 104
- slice\_type (*psd\_tools.psd.image\_resources.SliceV6* attribute), 104
- Slices (*class in psd\_tools.psd.image\_resources*), 104
- SLICES (*psd\_tools.constants.Resource* attribute), 80
- SlicesV6 (*class in psd\_tools.psd.image\_resources*), 104
- SliceV6 (*class in psd\_tools.psd.image\_resources*), 104
- SlopeLimitMatte (*psd\_tools.terminology.Enum* attribute), 140
- Small (*psd\_tools.terminology.Enum* attribute), 140
- SmallFontType (*psd\_tools.terminology.Key* attribute), 169
- smart\_object (*psd\_tools.api.layers.SmartObjectLayer* attribute), 60
- SMART\_OBJECT\_LAYER\_DATA1 (*psd\_tools.constants.Tag* attribute), 83
- SMART\_OBJECT\_LAYER\_DATA2 (*psd\_tools.constants.Tag* attribute), 83
- SmartBlur (*psd\_tools.terminology.Event* attribute), 149
- SmartBlurMode (*psd\_tools.terminology.Key* attribute), 169
- SmartBlurMode (*psd\_tools.terminology.Type* attribute), 177
- SmartBlurModeEdgeOnly (*psd\_tools.terminology.Enum* attribute), 140
- SmartBlurModeNormal (*psd\_tools.terminology.Enum* attribute), 140
- SmartBlurModeOverlayEdge (*psd\_tools.terminology.Enum* attribute), 140
- SmartBlurQuality (*psd\_tools.terminology.Key* attribute), 169
- SmartBlurQuality (*psd\_tools.terminology.Type* attribute), 177
- SmartBlurQualityHigh (*psd\_tools.terminology.Enum* attribute), 140
- SmartBlurQualityLow (*psd\_tools.terminology.Enum* attribute), 140
- SmartBlurQualityMedium (*psd\_tools.terminology.Enum* attribute), 140

- SmartObject (class in *psd\_tools.api.smart\_object*), 72
- SmartObjectLayer (class in *psd\_tools.api.layers*), 57
- SmartObjectLayerData (class in *psd\_tools.psd.tagged\_blocks*), 117
- Smooth (*psd\_tools.terminology.Event attribute*), 149
- Smooth (*psd\_tools.terminology.Key attribute*), 169
- Smoothness (*psd\_tools.terminology.Key attribute*), 169
- SMPTEC (*psd\_tools.terminology.Enum attribute*), 139
- SmudgeStick (*psd\_tools.terminology.Event attribute*), 149
- SmudgeTool (*psd\_tools.terminology.Klass attribute*), 125
- Snapshot (*psd\_tools.terminology.Enum attribute*), 140
- Snapshot (*psd\_tools.terminology.Klass attribute*), 125
- SnapshotInitial (*psd\_tools.terminology.Key attribute*), 169
- SOFT\_LIGHT (*psd\_tools.constants.BlendMode attribute*), 74
- SoftClip (*psd\_tools.terminology.Key attribute*), 169
- soften (*psd\_tools.api.effects.BevelEmboss attribute*), 40
- SoftLight (*psd\_tools.terminology.Enum attribute*), 140
- SoftMatte (*psd\_tools.terminology.Enum attribute*), 140
- Softness (*psd\_tools.terminology.Key attribute*), 169
- Solarize (*psd\_tools.terminology.Event attribute*), 149
- SOLID\_COLOR\_SHEET\_SETTING (*psd\_tools.constants.Tag attribute*), 83
- SOLID\_FILL (*psd\_tools.constants.EffectOSType attribute*), 76
- SolidColor (*psd\_tools.terminology.Enum attribute*), 140
- SolidColorFill (class in *psd\_tools.api.adjustments*), 17
- SolidFill (*psd\_tools.terminology.Key attribute*), 169
- SolidFill (*psd\_tools.terminology.Klass attribute*), 125
- SolidFillInfo (class in *psd\_tools.psd.effects\_layer*), 96
- Source (*psd\_tools.terminology.Key attribute*), 169
- Source2 (*psd\_tools.terminology.Key attribute*), 169
- SourceMode (*psd\_tools.terminology.Key attribute*), 169
- SourceMode (*psd\_tools.terminology.Type attribute*), 177
- Spacing (*psd\_tools.terminology.Key attribute*), 169
- Spatter (*psd\_tools.terminology.Event attribute*), 149
- SpecialInstructions (*psd\_tools.terminology.Key attribute*), 169
- Spectrum (*psd\_tools.terminology.Enum attribute*), 140
- Spherize (*psd\_tools.terminology.Event attribute*), 149
- SpherizeMode (*psd\_tools.terminology.Key attribute*), 169
- SpherizeMode (*psd\_tools.terminology.Type attribute*), 177
- Spin (*psd\_tools.terminology.Enum attribute*), 140
- SplitChannels (*psd\_tools.terminology.Event attribute*), 149
- Sponge (*psd\_tools.terminology.Event attribute*), 149
- Spot (*psd\_tools.terminology.Key attribute*), 169
- SPOT\_HALFTONE (*psd\_tools.constants.Resource attribute*), 80
- SpotColor (*psd\_tools.terminology.Enum attribute*), 140
- SpotColorChannel (*psd\_tools.terminology.Klass attribute*), 125
- SprayedStrokes (*psd\_tools.terminology.Event attribute*), 149
- SprayRadius (*psd\_tools.terminology.Key attribute*), 170
- Square (*psd\_tools.terminology.Enum attribute*), 140
- SquareSize (*psd\_tools.terminology.Key attribute*), 170
- SrcBlackMax (*psd\_tools.terminology.Key attribute*), 170
- SrcBlackMin (*psd\_tools.terminology.Key attribute*), 170
- SrcWhiteMax (*psd\_tools.terminology.Key attribute*), 170
- SrcWhiteMin (*psd\_tools.terminology.Key attribute*), 170
- SRGB (*psd\_tools.terminology.Enum attribute*), 139
- Stagger (*psd\_tools.terminology.Enum attribute*), 140
- StainedGlass (*psd\_tools.terminology.Event attribute*), 150
- Stamp (*psd\_tools.terminology.Event attribute*), 150
- StampIn (*psd\_tools.terminology.Enum attribute*), 140
- StampOut (*psd\_tools.terminology.Enum attribute*), 140
- Standard (*psd\_tools.terminology.Enum attribute*), 140
- Start (*psd\_tools.terminology.Key attribute*), 170
- StartArrowhead (*psd\_tools.terminology.Key attribute*), 170
- State (*psd\_tools.terminology.Key attribute*), 170
- State (*psd\_tools.terminology.Type attribute*), 177
- StdA (*psd\_tools.terminology.Enum attribute*), 140
- StdB (*psd\_tools.terminology.Enum attribute*), 140
- StdC (*psd\_tools.terminology.Enum attribute*), 140
- StdE (*psd\_tools.terminology.Enum attribute*), 140
- Stop (*psd\_tools.terminology.Event attribute*), 150
- Strength (*psd\_tools.terminology.Key attribute*), 170
- Strength\_PLUGIN (*psd\_tools.terminology.Key attribute*), 170
- StrengthRatio (*psd\_tools.terminology.Key attribute*), 170

- StretchToFit (*psd\_tools.terminology.Enum attribute*), 140
- String (*class in psd\_tools.psd.descriptor*), 91
- String (*class in psd\_tools.psd.engine\_data*), 93
- StringChannel (*psd\_tools.terminology.Type attribute*), 177
- StringClassFormat (*psd\_tools.terminology.Type attribute*), 177
- StringCompensation (*psd\_tools.terminology.Type attribute*), 177
- StringElement (*class in psd\_tools.psd.base*), 86
- StringFSS (*psd\_tools.terminology.Type attribute*), 177
- StringInteger (*psd\_tools.terminology.Type attribute*), 177
- Stroke (*class in psd\_tools.api.effects*), 39
- Stroke (*class in psd\_tools.api.shape*), 68
- stroke (*psd\_tools.api.adjustments.GradientFill attribute*), 28
- stroke (*psd\_tools.api.adjustments.PatternFill attribute*), 24
- stroke (*psd\_tools.api.adjustments.SolidColorFill attribute*), 20
- stroke (*psd\_tools.api.layers.Artboard attribute*), 45
- stroke (*psd\_tools.api.layers.Group attribute*), 49
- stroke (*psd\_tools.api.layers.PixelLayer attribute*), 53
- stroke (*psd\_tools.api.layers.ShapeLayer attribute*), 56
- stroke (*psd\_tools.api.layers.SmartObjectLayer attribute*), 60
- stroke (*psd\_tools.api.layers.TypeLayer attribute*), 65
- Stroke (*psd\_tools.terminology.Event attribute*), 150
- stroke\_adjust (*psd\_tools.api.shape.Stroke attribute*), 69
- StrokeDetail (*psd\_tools.terminology.Key attribute*), 170
- StrokeDirection (*psd\_tools.terminology.Key attribute*), 170
- StrokeDirection (*psd\_tools.terminology.Type attribute*), 177
- StrokeDirHorizontal (*psd\_tools.terminology.Enum attribute*), 140
- StrokeDirLeftDiag (*psd\_tools.terminology.Enum attribute*), 141
- StrokeDirRightDiag (*psd\_tools.terminology.Enum attribute*), 141
- StrokeDirVertical (*psd\_tools.terminology.Enum attribute*), 141
- StrokeLength (*psd\_tools.terminology.Key attribute*), 170
- StrokeLocation (*psd\_tools.terminology.Type attribute*), 177
- StrokePressure (*psd\_tools.terminology.Key attribute*), 170
- StrokeSize (*psd\_tools.terminology.Key attribute*), 170
- StrokeWidth (*psd\_tools.terminology.Key attribute*), 170
- style (*psd\_tools.psd.image\_resources.PrintScale attribute*), 103
- Style (*psd\_tools.terminology.Key attribute*), 170
- Style (*psd\_tools.terminology.Klass attribute*), 125
- Styles (*psd\_tools.terminology.Key attribute*), 170
- StylesAppend (*psd\_tools.terminology.Enum attribute*), 141
- StylesDelete (*psd\_tools.terminology.Enum attribute*), 141
- StylesLoad (*psd\_tools.terminology.Enum attribute*), 141
- StylesNew (*psd\_tools.terminology.Enum attribute*), 141
- StylesReset (*psd\_tools.terminology.Enum attribute*), 141
- StylesSave (*psd\_tools.terminology.Enum attribute*), 141
- StylusIsColor (*psd\_tools.terminology.Key attribute*), 170
- StylusIsOpacity (*psd\_tools.terminology.Key attribute*), 170
- StylusIsPressure (*psd\_tools.terminology.Key attribute*), 170
- StylusIsSize (*psd\_tools.terminology.Key attribute*), 170
- sub\_type (*psd\_tools.psd.tagged\_blocks.SectionDividerSetting attribute*), 117
- Subpath (*class in psd\_tools.psd.vector*), 119
- SubPath (*psd\_tools.terminology.Klass attribute*), 126
- SubPathList (*psd\_tools.terminology.Key attribute*), 170
- SUBTRACT (*psd\_tools.constants.BlendMode attribute*), 74
- Subtract (*psd\_tools.terminology.Enum attribute*), 141
- Subtract (*psd\_tools.terminology.Event attribute*), 150
- SubtractFrom (*psd\_tools.terminology.Event attribute*), 150
- Sumie (*psd\_tools.terminology.Event attribute*), 150
- SupplementalCategories (*psd\_tools.terminology.Key attribute*), 170
- SwatchesAppend (*psd\_tools.terminology.Enum attribute*), 141
- SwatchesReplace (*psd\_tools.terminology.Enum attribute*), 141
- SwatchesReset (*psd\_tools.terminology.Enum attribute*), 141
- SwatchesSave (*psd\_tools.terminology.Enum attribute*), 141
- SystemInfo (*psd\_tools.terminology.Key attribute*), 170
- SystemPalette (*psd\_tools.terminology.Key attribute*), 170

- tribute), 170
- SystemPicker (*psd\_tools.terminology.Enum attribute*), 141
- ## T
- Tables (*psd\_tools.terminology.Enum attribute*), 141
- Tag (*class in psd\_tools.constants*), 81
- tagged\_blocks (*psd\_tools.api.adjustments.GradientFill attribute*), 28
- tagged\_blocks (*psd\_tools.api.adjustments.PatternFill attribute*), 24
- tagged\_blocks (*psd\_tools.api.adjustments.SolidColorFill attribute*), 20
- tagged\_blocks (*psd\_tools.api.layers.Artboard attribute*), 45
- tagged\_blocks (*psd\_tools.api.layers.Group attribute*), 49
- tagged\_blocks (*psd\_tools.api.layers.PixelLayer attribute*), 53
- tagged\_blocks (*psd\_tools.api.layers.ShapeLayer attribute*), 56
- tagged\_blocks (*psd\_tools.api.layers.SmartObjectLayer attribute*), 60
- tagged\_blocks (*psd\_tools.api.layers.TypeLayer attribute*), 65
- tagged\_blocks (*psd\_tools.psd.layer\_and\_mask.LayerAndMaskInformation attribute*), 107
- tagged\_blocks (*psd\_tools.psd.layer\_and\_mask.LayerRecord attribute*), 108
- tagged\_blocks (*psd\_tools.PSDImage attribute*), 15
- TaggedBlock (*class in psd\_tools.psd.tagged\_blocks*), 115
- TaggedBlocks (*class in psd\_tools.psd.tagged\_blocks*), 114
- TakeMergedSnapshot (*psd\_tools.terminology.Event attribute*), 150
- TakeSnapshot (*psd\_tools.terminology.Event attribute*), 150
- TargaFormat (*psd\_tools.terminology.Klass attribute*), 126
- target (*psd\_tools.psd.image\_resources.SliceV6 attribute*), 104
- Target (*psd\_tools.terminology.Enum attribute*), 141
- Target (*psd\_tools.terminology.Key attribute*), 170
- TargetPath (*psd\_tools.terminology.Enum attribute*), 141
- TargetPath (*psd\_tools.terminology.Key attribute*), 170
- TargetPathIndex (*psd\_tools.terminology.Key attribute*), 170
- TermLength (*psd\_tools.terminology.Key attribute*), 170
- text (*psd\_tools.api.layers.TypeLayer attribute*), 65
- Text (*psd\_tools.terminology.Key attribute*), 170
- text\_data (*psd\_tools.psd.tagged\_blocks.TypeToolObjectSetting attribute*), 118
- TEXT\_ENGINE\_DATA (*psd\_tools.constants.Tag attribute*), 83
- text\_version (*psd\_tools.psd.tagged\_blocks.TypeToolObjectSetting attribute*), 118
- TextClickPoint (*psd\_tools.terminology.Key attribute*), 170
- TextData (*psd\_tools.terminology.Key attribute*), 170
- TextLayer (*psd\_tools.terminology.Klass attribute*), 126
- TextStyle (*psd\_tools.terminology.Key attribute*), 170
- TextStyle (*psd\_tools.terminology.Klass attribute*), 126
- TextStyleRange (*psd\_tools.terminology.Key attribute*), 171
- TextStyleRange (*psd\_tools.terminology.Klass attribute*), 126
- Texture (*psd\_tools.terminology.Key attribute*), 171
- TextureCoverage (*psd\_tools.terminology.Key attribute*), 171
- TextureFile (*psd\_tools.terminology.Key attribute*), 171
- TextureFill (*psd\_tools.terminology.Event attribute*), 150
- TextureType (*psd\_tools.terminology.Key attribute*), 171
- TextureType (*psd\_tools.terminology.Type attribute*), 177
- Texturizer (*psd\_tools.terminology.Event attribute*), 150
- TexTypeBlocks (*psd\_tools.terminology.Enum attribute*), 141
- TexTypeBrick (*psd\_tools.terminology.Enum attribute*), 141
- TexTypeBurlap (*psd\_tools.terminology.Enum attribute*), 141
- TexTypeCanvas (*psd\_tools.terminology.Enum attribute*), 141
- TexTypeFrosted (*psd\_tools.terminology.Enum attribute*), 141
- TexTypeSandstone (*psd\_tools.terminology.Enum attribute*), 141
- TexTypeTinyLens (*psd\_tools.terminology.Enum attribute*), 141
- Threshold (*class in psd\_tools.api.adjustments*), 33
- threshold (*psd\_tools.api.adjustments.Threshold attribute*), 33
- THRESHOLD (*psd\_tools.constants.Tag attribute*), 83
- Threshold (*psd\_tools.terminology.Enum attribute*), 141
- Threshold (*psd\_tools.terminology.Event attribute*), 150
- Threshold (*psd\_tools.terminology.Key attribute*), 171



Threshold (*psd\_tools.terminology.Klass attribute*), 126  
 Thumbnail (*psd\_tools.terminology.Enum attribute*), 141  
 thumbnail () (*psd\_tools.PSDImage method*), 15  
 THUMBNAIL\_RESOURCE (*psd\_tools.constants.Resource attribute*), 80  
 THUMBNAIL\_RESOURCE\_PS4 (*psd\_tools.constants.Resource attribute*), 80  
 ThumbnailResource (class in *psd\_tools.psd.image\_resources*), 105  
 ThumbnailResourceV4 (class in *psd\_tools.psd.image\_resources*), 105  
 TIFF (*psd\_tools.terminology.Enum attribute*), 141  
 TIFFFormat (*psd\_tools.terminology.Klass attribute*), 126  
 Tile (*psd\_tools.terminology.Enum attribute*), 141  
 Tile\_PLUGIN (*psd\_tools.terminology.Enum attribute*), 141  
 TileNumber (*psd\_tools.terminology.Key attribute*), 171  
 TileOffset (*psd\_tools.terminology.Key attribute*), 171  
 Tiles (*psd\_tools.terminology.Event attribute*), 150  
 TileSize (*psd\_tools.terminology.Key attribute*), 171  
 TIMELINE\_INFO (*psd\_tools.constants.Resource attribute*), 81  
 timestamp (*psd\_tools.psd.linked\_layer.LinkedLayer attribute*), 113  
 tint\_color (*psd\_tools.api.adjustments.BlackAndWhite attribute*), 32  
 Title (*psd\_tools.terminology.Key attribute*), 171  
 To (*psd\_tools.terminology.Key attribute*), 171  
 ToBuiltin (*psd\_tools.terminology.Key attribute*), 171  
 tobytes () (*psd\_tools.psd.base.BaseElement method*), 85  
 ToggleActionsPalette (*psd\_tools.terminology.Enum attribute*), 141  
 ToggleBlackPreview (*psd\_tools.terminology.Enum attribute*), 141  
 ToggleBrushesPalette (*psd\_tools.terminology.Enum attribute*), 141  
 ToggleChannelsPalette (*psd\_tools.terminology.Enum attribute*), 141  
 ToggleCMYKPreview (*psd\_tools.terminology.Enum attribute*), 141  
 ToggleCMYPreview (*psd\_tools.terminology.Enum attribute*), 141  
 ToggleColorPalette (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleCyanPreview (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleDocumentPalette (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleEdges (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleGamutWarning (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleGrid (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleGuides (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleHistoryPalette (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleInfoPalette (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleLayerMask (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleLayersPalette (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleLockGuides (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleMagentaPreview (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleNavigatorPalette (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleOptionsPalette (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleOthers (*psd\_tools.terminology.Key attribute*), 171  
 TogglePaths (*psd\_tools.terminology.Enum attribute*), 142  
 TogglePathsPalette (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleRGBMacPreview (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleRGBUncompensatedPreview (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleRGBWindowsPreview (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleRulers (*psd\_tools.terminology.Enum attribute*), 142  
 ToggleSnapToGrid (*psd\_tools.terminology.Enum attribute*), 142

- ToggleSnapToGuides (*psd\_tools.terminology.Enum attribute*), 142
- ToggleStatusBar (*psd\_tools.terminology.Enum attribute*), 142
- ToggleStylesPalette (*psd\_tools.terminology.Enum attribute*), 142
- ToggleSwatchesPalette (*psd\_tools.terminology.Enum attribute*), 142
- ToggleToolsPalette (*psd\_tools.terminology.Enum attribute*), 142
- ToggleYellowPreview (*psd\_tools.terminology.Enum attribute*), 142
- Tolerance (*psd\_tools.terminology.Key attribute*), 171
- ToLinked (*psd\_tools.terminology.Key attribute*), 171
- ToMode (*psd\_tools.terminology.Key attribute*), 171
- Tool (*psd\_tools.terminology.Klass attribute*), 126
- top (*psd\_tools.api.adjustments.GradientFill attribute*), 28
- top (*psd\_tools.api.adjustments.PatternFill attribute*), 24
- top (*psd\_tools.api.adjustments.SolidColorFill attribute*), 20
- top (*psd\_tools.api.layers.Artboard attribute*), 45
- top (*psd\_tools.api.layers.PixelLayer attribute*), 53
- top (*psd\_tools.api.layers.ShapeLayer attribute*), 57
- top (*psd\_tools.api.layers.SmartObjectLayer attribute*), 61
- top (*psd\_tools.api.layers.TypeLayer attribute*), 65
- top (*psd\_tools.api.mask.Mask attribute*), 67
- top (*psd\_tools.psd.layer\_and\_mask.LayerRecord attribute*), 108
- top (*psd\_tools.psd.layer\_and\_mask.MaskData attribute*), 109
- top (*psd\_tools.psd.tagged\_blocks.TypeToolObjectSetting attribute*), 118
- top (*psd\_tools.psd.vector.ClipboardRecord attribute*), 119
- top (*psd\_tools.PSDImage attribute*), 15
- Top (*psd\_tools.terminology.Enum attribute*), 142
- Top (*psd\_tools.terminology.Key attribute*), 171
- topil () (*psd\_tools.api.adjustments.GradientFill method*), 28
- topil () (*psd\_tools.api.adjustments.PatternFill method*), 24
- topil () (*psd\_tools.api.adjustments.SolidColorFill method*), 20
- topil () (*psd\_tools.api.layers.Artboard method*), 45
- topil () (*psd\_tools.api.layers.Group method*), 49
- topil () (*psd\_tools.api.layers.PixelLayer method*), 53
- topil () (*psd\_tools.api.layers.ShapeLayer method*), 57
- topil () (*psd\_tools.api.layers.SmartObjectLayer method*), 61
- topil () (*psd\_tools.api.layers.TypeLayer method*), 65
- topil () (*psd\_tools.api.mask.Mask method*), 67
- topil () (*psd\_tools.psd.image\_resources.ThumbnailResource method*), 105
- topil () (*psd\_tools.PSDImage method*), 15
- TornEdges (*psd\_tools.terminology.Event attribute*), 150
- total\_size (*psd\_tools.psd.image\_resources.ThumbnailResource attribute*), 105
- TotalLimit (*psd\_tools.terminology.Key attribute*), 171
- TraceContour (*psd\_tools.terminology.Event attribute*), 150
- Tracking (*psd\_tools.terminology.Key attribute*), 171
- TransferFunction (*class in psd\_tools.psd.image\_resources*), 106
- TransferFunction (*psd\_tools.terminology.Key attribute*), 171
- TransferFunctions (*class in psd\_tools.psd.image\_resources*), 105
- TransferPoint (*psd\_tools.terminology.Klass attribute*), 126
- TransferSpec (*psd\_tools.terminology.Key attribute*), 171
- TransferSpec (*psd\_tools.terminology.Klass attribute*), 126
- transform (*psd\_tools.api.layers.TypeLayer attribute*), 66
- transform (*psd\_tools.psd.tagged\_blocks.TypeToolObjectSetting attribute*), 118
- Transform (*psd\_tools.terminology.Event attribute*), 150
- Transparency (*psd\_tools.terminology.Enum attribute*), 142
- Transparency (*psd\_tools.terminology.Key attribute*), 171
- TRANSPARENCY\_INDEX (*psd\_tools.constants.Resource attribute*), 81
- TRANSPARENCY\_MASK (*psd\_tools.constants.ChannelID attribute*), 74
- transparency\_protected (*psd\_tools.psd.layer\_and\_mask.LayerFlags attribute*), 109
- TRANSPARENCY\_SHAPES\_LAYER (*psd\_tools.constants.Tag attribute*), 83
- transparency\_stops (*psd\_tools.api.adjustments.GradientMap attribute*), 33
- TransparencyGamutPreferences (*psd\_tools.terminology.Enum attribute*), 142
- TransparencyGrid (*psd\_tools.terminology.Key at-*

- tribute*), 171
  - TransparencyGridColors (*psd\_tools.terminology.Key attribute*), 171
  - TransparencyGridColors (*psd\_tools.terminology.Type attribute*), 177
  - TransparencyGridSize (*psd\_tools.terminology.Key attribute*), 171
  - TransparencyGridSize (*psd\_tools.terminology.Type attribute*), 177
  - TransparencyPrefs (*psd\_tools.terminology.Key attribute*), 171
  - TransparencyPrefs (*psd\_tools.terminology.Klass attribute*), 126
  - TransparencyShape (*psd\_tools.terminology.Key attribute*), 171
  - TransparencyStop (*psd\_tools.terminology.Klass attribute*), 126
  - Transparent (*psd\_tools.terminology.Enum attribute*), 142
  - TransparentIndex (*psd\_tools.terminology.Key attribute*), 171
  - TransparentWhites (*psd\_tools.terminology.Key attribute*), 171
  - Trap (*psd\_tools.terminology.Event attribute*), 150
  - Trinitron (*psd\_tools.terminology.Enum attribute*), 142
  - Tritone (*psd\_tools.terminology.Enum attribute*), 142
  - Twirl (*psd\_tools.terminology.Event attribute*), 150
  - Twist (*psd\_tools.terminology.Key attribute*), 171
  - Type (*class in psd\_tools.terminology*), 173
  - type (*psd\_tools.api.effects.GradientOverlay attribute*), 38
  - Type (*psd\_tools.terminology.Key attribute*), 171
  - TYPE\_TOOL\_INFO (*psd\_tools.constants.Tag attribute*), 83
  - TYPE\_TOOL\_OBJECT\_SETTING (*psd\_tools.constants.Tag attribute*), 83
  - TypeClassModeOrClassMode (*psd\_tools.terminology.Type attribute*), 177
  - TypeID (*psd\_tools.psd.descriptor.Enumerated attribute*), 88
  - TypeID (*psd\_tools.psd.descriptor.EnumeratedReference attribute*), 89
  - TypeLayer (*class in psd\_tools.api.layers*), 61
  - TypeToolObjectSetting (*class in psd\_tools.psd.tagged\_blocks*), 118
- ## U
- UCA (*psd\_tools.terminology.Key attribute*), 171
  - UIBitmap (*psd\_tools.terminology.Enum attribute*), 142
  - UICMYK (*psd\_tools.terminology.Enum attribute*), 142
  - UIDuotone (*psd\_tools.terminology.Enum attribute*), 143
  - UIGrayscale (*psd\_tools.terminology.Enum attribute*), 143
  - UIIndexed (*psd\_tools.terminology.Enum attribute*), 143
  - UILab (*psd\_tools.terminology.Enum attribute*), 143
  - UIMultichannel (*psd\_tools.terminology.Enum attribute*), 143
  - UIRGB (*psd\_tools.terminology.Enum attribute*), 143
  - UndefinedArea (*psd\_tools.terminology.Key attribute*), 171
  - UndefinedArea (*psd\_tools.terminology.Type attribute*), 177
  - Underline (*psd\_tools.terminology.Key attribute*), 171
  - Underpainting (*psd\_tools.terminology.Event attribute*), 150
  - Undo (*psd\_tools.terminology.Enum attribute*), 143
  - Undo (*psd\_tools.terminology.Event attribute*), 150
  - Ungroup (*psd\_tools.terminology.Event attribute*), 150
  - UNICODE\_LAYER\_NAME (*psd\_tools.constants.Tag attribute*), 83
  - UNICODE\_PATH\_NAME (*psd\_tools.constants.Tag attribute*), 83
  - Uniform (*psd\_tools.terminology.Enum attribute*), 143
  - UniformDistribution (*psd\_tools.terminology.Enum attribute*), 143
  - unique\_id (*psd\_tools.api.smart\_object.SmartObject attribute*), 73
  - Unit (*class in psd\_tools.terminology*), 178
  - unit (*psd\_tools.psd.descriptor.UnitFloat attribute*), 91
  - unit (*psd\_tools.psd.descriptor.UnitFloats attribute*), 92
  - unit (*psd\_tools.psd.image\_resources.HalftoneScreen attribute*), 101
  - UnitFloat (*class in psd\_tools.psd.descriptor*), 91
  - UnitFloat (*psd\_tools.terminology.Type attribute*), 177
  - UnitFloats (*class in psd\_tools.psd.descriptor*), 92
  - UnitsPrefs (*psd\_tools.terminology.Key attribute*), 171
  - UnitsPrefs (*psd\_tools.terminology.Klass attribute*), 126
  - UnitsRulersPreferences (*psd\_tools.terminology.Enum attribute*), 143
  - UNKNOWN (*psd\_tools.constants.PlacedLayerType attribute*), 77
  - Unlink (*psd\_tools.terminology.Event attribute*), 150
  - UnsharpMask (*psd\_tools.terminology.Event attribute*), 150
  - UnspecifiedColor (*psd\_tools.terminology.Klass attribute*), 126
  - Untitled (*psd\_tools.terminology.Key attribute*), 172
  - Upper (*psd\_tools.terminology.Enum attribute*), 143
  - UpperY (*psd\_tools.terminology.Key attribute*), 172
  - Urgency (*psd\_tools.terminology.Key attribute*), 172

- Urgency (*psd\_tools.terminology.Type* attribute), 177
- URL (*psd\_tools.constants.Resource* attribute), 81
- url (*psd\_tools.psd.image\_resources.SliceV6* attribute), 104
- URL (*psd\_tools.terminology.Key* attribute), 171
- URL\_LIST (*psd\_tools.constants.Resource* attribute), 81
- URLItem (*class in psd\_tools.psd.image\_resources*), 106
- URLList (*class in psd\_tools.psd.image\_resources*), 106
- use\_accurate (*psd\_tools.psd.image\_resources.HalftoneScreen* attribute), 172
- use\_global\_angle (*psd\_tools.psd.effects\_layer.BevelInfo* attribute), 96
- use\_global\_angle (*psd\_tools.psd.effects\_layer.ShadowInfo* attribute), 94
- use\_global\_light (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- use\_global\_light (*psd\_tools.api.effects.DropShadow* attribute), 35
- use\_global\_light (*psd\_tools.api.effects.InnerShadow* attribute), 35
- use\_legacy (*psd\_tools.api.adjustments.BrightnessContrast* attribute), 29
- use\_printer (*psd\_tools.psd.image\_resources.HalftoneScreen* attribute), 102
- use\_shape (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- use\_texture (*psd\_tools.api.effects.BevelEmboss* attribute), 40
- use\_tint (*psd\_tools.api.adjustments.BlackAndWhite* attribute), 32
- use\_vector\_color (*psd\_tools.api.adjustments.GradientMap* attribute), 33
- UseAccurateScreens (*psd\_tools.terminology.Key* attribute), 172
- UseAdditionalPlugins (*psd\_tools.terminology.Key* attribute), 172
- UseCacheForHistograms (*psd\_tools.terminology.Key* attribute), 172
- UseCurves (*psd\_tools.terminology.Key* attribute), 172
- UseDefault (*psd\_tools.terminology.Key* attribute), 172
- UseGlobalAngle (*psd\_tools.terminology.Key* attribute), 172
- UseICCPProfile (*psd\_tools.terminology.Key* attribute), 172
- UseMask (*psd\_tools.terminology.Key* attribute), 172
- USER\_DEFINED (*psd\_tools.constants.PrintScaleStyle* attribute), 77
- USER\_LAYER\_MASK (*psd\_tools.constants.ChannelID* attribute), 74
- USER\_MASK (*psd\_tools.constants.Tag* attribute), 83
- user\_mask\_density (*psd\_tools.psd.layer\_and\_mask.MaskParameters* attribute), 111
- user\_mask\_feather (*psd\_tools.psd.layer\_and\_mask.MaskParameters* attribute), 111
- user\_mask\_from\_render (*psd\_tools.psd.layer\_and\_mask.MaskFlags* attribute), 110
- UserMask (*class in psd\_tools.psd.tagged\_blocks*), 118
- UserMaskEnabled (*psd\_tools.terminology.Key* attribute), 172
- UserMaskLinked (*psd\_tools.terminology.Key* attribute), 172
- UserMaskOptions (*psd\_tools.terminology.Type* attribute), 177
- UserStop (*psd\_tools.terminology.Enum* attribute), 143
- Using (*psd\_tools.terminology.Key* attribute), 172
- USING\_ALIGNED\_RENDERING (*psd\_tools.constants.Tag* attribute), 83
- uuid (*psd\_tools.psd.filter\_effects.FilterEffect* attribute), 97
- uuid (*psd\_tools.psd.linked\_layer.LinkedLayer* attribute), 113
- V**
- validate () (*psd\_tools.psd.base.BaseElement* method), 85
- value (*psd\_tools.psd.base.StringElement* attribute), 86
- value (*psd\_tools.psd.base.ValueElement* attribute), 85
- value (*psd\_tools.psd.descriptor.Bool* attribute), 87
- value (*psd\_tools.psd.descriptor.Double* attribute), 88
- value (*psd\_tools.psd.descriptor.Integer* attribute), 89
- value (*psd\_tools.psd.descriptor.LargeInteger* attribute), 89
- value (*psd\_tools.psd.descriptor.Name* attribute), 90
- value (*psd\_tools.psd.descriptor.Offset* attribute), 91
- value (*psd\_tools.psd.descriptor.RawData* attribute), 91
- value (*psd\_tools.psd.descriptor.String* attribute), 91
- value (*psd\_tools.psd.descriptor.UnitFloat* attribute), 91
- value (*psd\_tools.psd.tagged\_blocks.Bytes* attribute), 115
- value (*psd\_tools.psd.tagged\_blocks.SheetColorSetting* attribute), 117
- Value (*psd\_tools.terminology.Key* attribute), 172
- ValueElement (*class in psd\_tools.psd.base*), 85
- ValueList (*psd\_tools.terminology.Type* attribute), 177
- values (*psd\_tools.psd.descriptor.UnitFloats* attribute), 92
- Variance (*psd\_tools.terminology.Key* attribute), 172
- Variations (*psd\_tools.terminology.Event* attribute), 150
- VECTOR (*psd\_tools.constants.PlacedLayerType* attribute), 77
- Vector0 (*psd\_tools.terminology.Key* attribute), 172
- Vector1 (*psd\_tools.terminology.Key* attribute), 172



- `vector_mask` (`psd_tools.api.adjustments.GradientFill attribute`), 29
- `vector_mask` (`psd_tools.api.adjustments.PatternFill attribute`), 25
- `vector_mask` (`psd_tools.api.adjustments.SolidColorFill attribute`), 21
- `vector_mask` (`psd_tools.api.layers.Artboard attribute`), 45
- `vector_mask` (`psd_tools.api.layers.Group attribute`), 49
- `vector_mask` (`psd_tools.api.layers.PixelLayer attribute`), 53
- `vector_mask` (`psd_tools.api.layers.ShapeLayer attribute`), 57
- `vector_mask` (`psd_tools.api.layers.SmartObjectLayer attribute`), 61
- `vector_mask` (`psd_tools.api.layers.TypeLayer attribute`), 66
- `VECTOR_MASK_AS_GLOBAL_MASK` (`psd_tools.constants.Tag attribute`), 83
- `vector_mask_density` (`psd_tools.psd.layer_and_mask.MaskParameters attribute`), 111
- `vector_mask_feather` (`psd_tools.psd.layer_and_mask.MaskParameters attribute`), 111
- `VECTOR_MASK_SETTING1` (`psd_tools.constants.Tag attribute`), 84
- `VECTOR_MASK_SETTING2` (`psd_tools.constants.Tag attribute`), 84
- `VECTOR_ORIGINATION_DATA` (`psd_tools.constants.Tag attribute`), 84
- `VECTOR_STROKE_CONTENT_DATA` (`psd_tools.constants.Tag attribute`), 84
- `VECTOR_STROKE_DATA` (`psd_tools.constants.Tag attribute`), 84
- `VectorColor` (`psd_tools.terminology.Key attribute`), 172
- `VectorMask` (`class in psd_tools.api.shape`), 67
- `VectorMaskSetting` (`class in psd_tools.psd.vector`), 120
- `VectorStrokeContentSetting` (`class in psd_tools.psd.vector`), 120
- `version` (`psd_tools.psd.effects_layer.BevelInfo attribute`), 95
- `version` (`psd_tools.psd.effects_layer.CommonStateInfo attribute`), 94
- `version` (`psd_tools.psd.effects_layer.EffectsLayer attribute`), 94
- `version` (`psd_tools.psd.effects_layer.InnerGlowInfo attribute`), 95
- `version` (`psd_tools.psd.effects_layer.OuterGlowInfo attribute`), 94
- `version` (`psd_tools.psd.effects_layer.ShadowInfo attribute`), 94
- `version` (`psd_tools.psd.effects_layer.SolidFillInfo attribute`), 96
- `version` (`psd_tools.psd.filter_effects.FilterEffect attribute`), 97
- `version` (`psd_tools.psd.filter_effects.FilterEffects attribute`), 96
- `version` (`psd_tools.psd.header.FileHeader attribute`), 98
- `version` (`psd_tools.psd.image_resources.PrintFlagsInfo attribute`), 103
- `version` (`psd_tools.psd.image_resources.Slices attribute`), 104
- `version` (`psd_tools.psd.image_resources.VersionInfo attribute`), 106
- `version` (`psd_tools.psd.linked_layer.LinkedList attribute`), 112
- `version` (`psd_tools.psd.patterns.Pattern attribute`), 113
- `version` (`psd_tools.psd.patterns.VirtualMemoryArrayList attribute`), 114
- `version` (`psd_tools.psd.tagged_blocks.SmartObjectLayerData attribute`), 117
- `version` (`psd_tools.psd.tagged_blocks.TypeToolObjectSetting attribute`), 118
- `version` (`psd_tools.psd.vector.VectorMaskSetting attribute`), 120
- `version` (`psd_tools.psd.vector.VectorStrokeContentSetting attribute`), 120
- `version` (`psd_tools.PSDImage attribute`), 15
- `Version` (`psd_tools.terminology.Klass attribute`), 126
- `VERSION_INFO` (`psd_tools.constants.Resource attribute`), 81
- `VersionFix` (`psd_tools.terminology.Key attribute`), 172
- `VersionInfo` (`class in psd_tools.psd.image_resources`), 106
- `VersionMajor` (`psd_tools.terminology.Key attribute`), 172
- `VersionMinor` (`psd_tools.terminology.Key attribute`), 172
- `vertical` (`psd_tools.psd.image_resources.ResolutionInfo attribute`), 103
- `vertical` (`psd_tools.psd.image_resources.SliceV6 attribute`), 105
- `Vertical` (`psd_tools.terminology.Enum attribute`), 143
- `Vertical` (`psd_tools.terminology.Key attribute`), 172
- `vertical_unit` (`psd_tools.psd.image_resources.ResolutionInfo attribute`), 103
- `VerticalLocation` (`psd_tools.terminology.Type attribute`), 177
- `VerticalOnly` (`psd_tools.terminology.Enum attribute`), 143
- `VerticalScale` (`psd_tools.terminology.Key attribute`), 172

- Vibrance (*class in psd\_tools.api.adjustments*), 30  
vibrance (*psd\_tools.api.adjustments.Vibrance attribute*), 30  
VIBRANCE (*psd\_tools.constants.Tag attribute*), 84  
VideoAlpha (*psd\_tools.terminology.Key attribute*), 172  
viewbox (*psd\_tools.PSDImage attribute*), 15  
Violet (*psd\_tools.terminology.Enum attribute*), 143  
VirtualMemoryArray (*class in psd\_tools.psd.patterns*), 114  
VirtualMemoryArrayList (*class in psd\_tools.psd.patterns*), 114  
visible (*psd\_tools.api.adjustments.GradientFill attribute*), 29  
visible (*psd\_tools.api.adjustments.PatternFill attribute*), 25  
visible (*psd\_tools.api.adjustments.SolidColorFill attribute*), 21  
visible (*psd\_tools.api.layers.Artboard attribute*), 45  
visible (*psd\_tools.api.layers.Group attribute*), 49  
visible (*psd\_tools.api.layers.PixelLayer attribute*), 53  
visible (*psd\_tools.api.layers.ShapeLayer attribute*), 57  
visible (*psd\_tools.api.layers.SmartObjectLayer attribute*), 61  
visible (*psd\_tools.api.layers.TypeLayer attribute*), 66  
visible (*psd\_tools.psd.effects\_layer.CommonStateInfo attribute*), 94  
visible (*psd\_tools.psd.layer\_and\_mask.LayerFlags attribute*), 109  
visible (*psd\_tools.PSDImage attribute*), 16  
Visible (*psd\_tools.terminology.Key attribute*), 172  
VIVID\_LIGHT (*psd\_tools.constants.BlendMode attribute*), 74  
VMPreferences (*psd\_tools.terminology.Enum attribute*), 143  
vrnsn (*psd\_tools.api.adjustments.BrightnessContrast attribute*), 29
- ## W
- Wait (*psd\_tools.terminology.Event attribute*), 150  
warp (*psd\_tools.api.layers.TypeLayer attribute*), 66  
warp (*psd\_tools.api.smart\_object.SmartObject attribute*), 73  
warp (*psd\_tools.psd.tagged\_blocks.TypeToolObjectSetting attribute*), 118  
warp\_version (*psd\_tools.psd.tagged\_blocks.TypeToolObjectSetting attribute*), 118  
WatchSuspension (*psd\_tools.terminology.Key attribute*), 172  
Watercolor (*psd\_tools.terminology.Event attribute*), 150  
WATERMARK (*psd\_tools.constants.Resource attribute*), 81  
Watermark (*psd\_tools.terminology.Key attribute*), 172  
WaterPaper (*psd\_tools.terminology.Event attribute*), 150  
Wave (*psd\_tools.terminology.Event attribute*), 150  
WavelengthMax (*psd\_tools.terminology.Key attribute*), 172  
WavelengthMin (*psd\_tools.terminology.Key attribute*), 172  
WaveSine (*psd\_tools.terminology.Enum attribute*), 143  
WaveSquare (*psd\_tools.terminology.Enum attribute*), 143  
WaveTriangle (*psd\_tools.terminology.Enum attribute*), 143  
WaveType (*psd\_tools.terminology.Key attribute*), 172  
WaveType (*psd\_tools.terminology.Type attribute*), 178  
Web (*psd\_tools.terminology.Enum attribute*), 143  
WebdavPrefs (*psd\_tools.terminology.Key attribute*), 172  
WebdavPrefs (*psd\_tools.terminology.Klass attribute*), 126  
WetEdges (*psd\_tools.terminology.Key attribute*), 172  
What (*psd\_tools.terminology.Key attribute*), 172  
White (*psd\_tools.terminology.Enum attribute*), 143  
WhiteClip (*psd\_tools.terminology.Key attribute*), 172  
WhiteIntensity (*psd\_tools.terminology.Key attribute*), 172  
WhiteIsHigh (*psd\_tools.terminology.Key attribute*), 173  
WhiteLevel (*psd\_tools.terminology.Key attribute*), 173  
WhitePoint (*psd\_tools.terminology.Key attribute*), 173  
Whites (*psd\_tools.terminology.Enum attribute*), 143  
WholePath (*psd\_tools.terminology.Key attribute*), 173  
WideGamutRGB (*psd\_tools.terminology.Enum attribute*), 143  
WidePhosphors (*psd\_tools.terminology.Enum attribute*), 143  
width (*psd\_tools.api.adjustments.GradientFill attribute*), 29  
width (*psd\_tools.api.adjustments.PatternFill attribute*), 25  
width (*psd\_tools.api.adjustments.SolidColorFill attribute*), 21  
width (*psd\_tools.api.layers.Artboard attribute*), 45  
width (*psd\_tools.api.layers.Group attribute*), 49  
width (*psd\_tools.api.layers.PixelLayer attribute*), 53  
width (*psd\_tools.api.layers.ShapeLayer attribute*), 57  
width (*psd\_tools.api.layers.SmartObjectLayer attribute*), 61  
width (*psd\_tools.api.layers.TypeLayer attribute*), 66  
width (*psd\_tools.api.mask.Mask attribute*), 67  
width (*psd\_tools.psd.header.FileHeader attribute*), 98  
width (*psd\_tools.psd.image\_resources.ThumbnailResource*

attribute), 105  
 width (*psd\_tools.psd.layer\_and\_mask.LayerRecord* attribute), 108  
 width (*psd\_tools.psd.layer\_and\_mask.MaskData* attribute), 110  
 width (*psd\_tools.PSDImage* attribute), 16  
 Width (*psd\_tools.terminology.Key* attribute), 173  
 width\_unit (*psd\_tools.psd.image\_resources.ResolutionInfo* attribute), 103  
 Wind (*psd\_tools.terminology.Enum* attribute), 143  
 Wind (*psd\_tools.terminology.Event* attribute), 150  
 WindMethod (*psd\_tools.terminology.Key* attribute), 173  
 WindMethod (*psd\_tools.terminology.Type* attribute), 178  
 Windows (*psd\_tools.terminology.Enum* attribute), 143  
 WINDOWS\_DEVMODE (*psd\_tools.constants.Resource* attribute), 81  
 WindowsSystem (*psd\_tools.terminology.Enum* attribute), 143  
 WinThumbnail (*psd\_tools.terminology.Enum* attribute), 143  
 With (*psd\_tools.terminology.Key* attribute), 173  
 WORKFLOW\_URL (*psd\_tools.constants.Resource* attribute), 81  
 WORKING\_PATH (*psd\_tools.constants.Resource* attribute), 81  
 WorkPath (*psd\_tools.terminology.Enum* attribute), 143  
 WorkPath (*psd\_tools.terminology.Key* attribute), 173  
 WorkPathIndex (*psd\_tools.terminology.Key* attribute), 173  
 Wrap (*psd\_tools.terminology.Enum* attribute), 143  
 WrapAround (*psd\_tools.terminology.Enum* attribute), 143  
 write() (*psd\_tools.psd.base.BaseElement* method), 85  
 writer (*psd\_tools.psd.image\_resources.VersionInfo* attribute), 106

## X

x (*psd\_tools.psd.image\_resources.PrintScale* attribute), 103  
 X (*psd\_tools.terminology.Key* attribute), 173  
 XMP\_METADATA (*psd\_tools.constants.Resource* attribute), 81  
 XYColor (*psd\_tools.terminology.Klass* attribute), 126  
 xyz (*psd\_tools.api.adjustments.PhotoFilter* attribute), 32

## Y

y (*psd\_tools.psd.image\_resources.PrintScale* attribute), 103  
 Y (*psd\_tools.terminology.Key* attribute), 173  
 yellow (*psd\_tools.api.adjustments.BlackAndWhite* attribute), 32  
 Yellow (*psd\_tools.terminology.Enum* attribute), 143

Yellow (*psd\_tools.terminology.Key* attribute), 173  
 YellowColor (*psd\_tools.terminology.Enum* attribute), 143  
 Yellows (*psd\_tools.terminology.Enum* attribute), 143  
 Yes (*psd\_tools.terminology.Enum* attribute), 143  
 YesNo (*psd\_tools.terminology.Type* attribute), 178

## Z

ZigZag (*psd\_tools.terminology.Event* attribute), 150  
 ZigZagType (*psd\_tools.terminology.Key* attribute), 173  
 ZigZagType (*psd\_tools.terminology.Type* attribute), 178  
 ZIP (*psd\_tools.constants.Compression* attribute), 75  
 Zip (*psd\_tools.terminology.Enum* attribute), 143  
 ZIP\_WITH\_PREDICTION (*psd\_tools.constants.Compression* attribute), 76  
 Zoom (*psd\_tools.terminology.Enum* attribute), 144  
 ZoomIn (*psd\_tools.terminology.Enum* attribute), 144  
 ZoomOut (*psd\_tools.terminology.Enum* attribute), 144